

More Tight Bounds for Active Self-Assembly Using an Insertion Primitive*

Benjamin Hescott[†] Caleb Malchik[‡] Andrew Winslow[§]

Abstract

We prove several limits on the behavior of a model of self-assembling particles introduced by Dabby and Chen (SODA 2013), called *insertion systems*, where monomers insert themselves into the middle of a growing linear polymer. First, we prove that the expressive power of these systems is equal to context-free grammars, answering a question posed by Dabby and Chen.

Second, we give tight bounds on the maximum length and minimum expected time of constructed polymers in systems of three increasingly restricted classes. We prove that systems of k monomer types can deterministically construct polymers of length $n = 2^{\Theta(k^{3/2})}$ in $O(\log^{5/3}(n))$ expected time. We also prove that if non-deterministic construction of a finite number of polymers is permitted, then the expected construction time can be reduced to $O(\log^{3/2}(n))$ at the trade-off of decreasing the length to $2^{\Theta(k)}$. If the system is allowed to construct an infinite number of polymers, then constructing polymers of unbounded length in $O(\log n)$ expected time is possible. We follow these positive results with a set of lower bounds proving that these are the best possible polymer lengths and expected construction times.

1 Introduction

In this work we study a theoretical model of *algorithmic self-assembly*, in which simple particles aggregate in a distributed manner to carry out complex functionality. Perhaps the the most well-studied theoretical model of algorithmic self-assembly is the *abstract Tile Assembly Model (aTAM)* of Winfree [23] consisting of square *tiles* irreversibly attach to a growing polyomino-shaped assembly according to matching edge colors. This model is capable of Turing-universal computation [23], self-simulation [8], and efficient assembly of general (scaled) shapes [22] and squares [1, 21]. Despite this power, the model is incapable of assembling some shapes efficiently; a single row of n tiles requires n distinct tile

*Some of the results in this work have been published as [18].

[†]Tufts University, Department of Computer Science, hescott@cs.tufts.edu

[‡]Tufts University, Department of Computer Science, caleb.malchik@tufts.edu

[§]Université Libre de Bruxelles, Département d'Informatique, awinslow@ulb.ac.be

types and $\Omega(n)$ expected assembly time [1] and any shape with n tiles requires $\Omega(\sqrt{n})$ expected time to assemble [13].

Such a limitation may not seem so significant, except that a wide range of biological systems form complex assemblies in time polylogarithmic in the assembly size, as Dabby and Chen [6] and Woods et al. [24] observe. These biological systems are capable of such growth because their particles (e.g. living cells) *actively* carry out geometric reconfiguration. In the interest of both understanding naturally occurring biological systems and creating synthetic systems with additional capabilities, several models of *active self-assembly* have been proposed recently. These include the graph grammars of Klavins et al. [15, 16], the *nubots* model of Woods et al. [3, 4, 24], and the insertion systems of Dabby and Chen [6]. Both graph grammars and nubots are capable of a topologically rich set of assemblies and reconfigurations, but rely on stateful particles forming complex bond arrangements. In contrast, insertion systems consist of stateless particles forming a single chain of bonds. Indeed, all insertion systems are captured as a special case of nubots in which a linear polymer is assembled via parallel insertion-like reconfigurations, as in Theorem 5.1 of [25]. The simplicity of insertion systems makes their implementation in matter a more immediately attainable goal; Dabby and Chen [5, 6] describe a direct implementation of these systems in DNA.

We are careful to make a distinction between *active self-assembly* where assemblies undergo reconfiguration, and *active tile self-assembly* [9, 10, 11, 12, 14, 17, 19, 20], where tile-based assemblies change their bond structure. Active self-assembly enables exponential assembly rates by enabling insertion of new particles throughout the assembly, while active tile self-assembly does not: assemblies formed consist of rigid tiles and the $\Omega(\sqrt{n})$ expected-time lower bound of Keenan, Schweller, Sherman, and Zhong [13] still applies.

1.1 Our results

We prove two types of results on the behavior of insertion systems. We start by considering what languages can be *expressed* by insertion systems, i.e. correspond to a set of polymers constructed by some insertion system. Dabby and Chen prove that only context-free languages are expressible by insertion systems, and ask whether every context-free language is indeed expressed by some insertion system. We answer this question in the affirmative, and as a consequence prove that the languages expressible by insertion systems are exactly the context-free languages.

After achieving a tight bound on the expressive power of insertion systems, we turn to considering the efficiency of insertion systems, both with regards to the number of monomer types used and the expected time to construct polymers. Dabby and Chen prove that insertion systems with k monomer types can deterministically construct polymers of length $n = 2^{\Theta(\sqrt{k})}$ in $O(\log^3 n)$ expected time. In Section 4 we describe three constructions in this vein. First, we improve on the result of Dabby and Chen, proving that deterministic construction

of polymers with length $n = 2^{\Theta(k^{3/2})}$ in $O(\log^{5/3}(n))$ expected time is possible (Theorem 4.2). Second, we prove that allowing non-deterministic construction of a finite set of polymers enables constructing polymers of length $n = 2^{\Theta(k)}$ in $O(\log^{3/2}(n))$ expected time (Theorem 4.3). Third, we briefly describe a 2-monomer-type system constructing polymers of all lengths $n \geq 3$ in $O(\log n)$ expected time, which can easily be seen to be optimal for unrestricted insertion systems.

In Section 5, we prove that these systems are each optimal with regards to both polymer length and expected construction time. First, we prove that deterministically constructing a polymer of length n takes $\Omega(\log^{5/3}(n))$ expected time (Theorem 5.4), matching the construction time of Theorem 4.2 and proving that no trade-off between monomer types and construction is possible for deterministic systems.

Next, we prove that constructing a polymer of length n in a system constructing a finite set of polymers, including deterministic systems, requires $\Omega(\log^{2/3}(n))$ monomer types and, if $\Theta(\log^{2/3}(n))$ types are used, $\Omega(\log^{5/3}(n))$ expected time (Theorem 5.6). Both of these bounds match those achieved by the construction of Theorem 4.2.

Finally, we prove that constructing a polymer of length n in a system constructing a finite set of polymers requires $\Omega(\log^{3/2}(n))$ expected time and, if $\Theta(\log^{3/2}(n))$ expected time is achieved, $\Omega(\log n)$ monomer types (Theorem 5.8). Again, both of these bounds match those achieved by the construction of Theorem 4.3.

Taken together, these results give an asymptotically tight characterization of maximum length and minimum expected time of polymer constructions for three general classes of insertion systems: deterministic construction, construction of a finite set of polymers, and unrestricted construction. Our lower bounds also imply a length and time tradeoff for systems constructing a finite set of polymers: constructing a polymer of length n using k monomer types takes $\Omega(\log^2(n)/\sqrt{k})$ (Lemma 5.5).

2 Definitions

Section 2.1 defines standard context-free grammars, as well as a special type called *pair grammars*, used in Section 3. Section 2.2 defines insertion systems, with a small number of modifications from the definitions given in [6] designed to ease readability. Section 2.3 formalizes the notion of expressive power used in [6].

2.1 Grammars

A *context-free grammar* \mathcal{G} is a 4-tuple $\mathcal{G} = (\Sigma, \Gamma, \Delta, S)$. The sets Σ and Γ are the *terminal* and *non-terminal symbols* of the grammar. The set Δ consists of *production rules* or simply *rules*, each of the form $L \rightarrow R_1 R_2 \cdots R_j$ with $L \in \Gamma$ and $R_i \in \Sigma \cup \Gamma$. Finally, the symbol $S \in \Gamma$ is a special *start symbol*. The

language of \mathcal{G} , denoted $L(\mathcal{G})$, is the set of finite strings that can be *derived* by starting with S , and repeatedly replacing a non-terminal symbol found on the left-hand side of some rule in Δ with the sequence of symbols on the right-hand side of the rule. The *size* of \mathcal{G} is $|\Delta|$, the number of rules in \mathcal{G} . If every rule in Δ is of the form $L \rightarrow R_1 R_2$ or $L \rightarrow t$, with $R_1 R_2 \in \Gamma$ and $t \in \Sigma$, then the grammar is said to be in *Chomsky normal form*.

An *integer-pair grammar*, used in Section 3, is a context-free grammar in Chomsky normal form such that each non-terminal symbol is an integer pair (a, d) , and each production rule has the form $(a, d) \rightarrow (a, b)(c, d)$ or $(a, d) \rightarrow t$.

2.2 Insertion systems

An *insertion system* in the active self-assembly model of Dabby and Chen [6] carries out the construction of a linear *polymer* consisting of constant length *monomers*. A polymer grows incrementally by the insertion of a monomer at an *insertion site* between two existing monomers in the polymer, according to complementary bonding sites between the monomer and the insertion site.

An insertion system \mathcal{S} is defined as a 4-tuple $\mathcal{S} = (\Sigma, \Delta, Q, R)$. The first element, Σ , is a set of symbols. Each symbol $s \in \Sigma$ has a *complement* s^* . We denote the complement of a symbol s as \bar{s} , i.e. $\bar{s} = s^*$ and $\overline{s^*} = s$. The set Δ is a set of *monomer types*, each assigned a *concentration*. Each monomer is specified by a quadruple $(a, b, c, d)^+$ or $(a, b, c, d)^-$, where $a, b, c, d \in \Sigma \cup \{s^* : s \in \Sigma\}$, and each concentration is a real number between 0 and 1. The sum of all concentrations in Δ must be at most 1. The two symbols $Q = (a, b)$ and $R = (c, d)$ are special two-symbol monomers that together form the *initiator* of \mathcal{S} . It is required that either $\bar{a} = d$ or $\bar{b} = c$. The *size* of \mathcal{S} is $|\Delta|$, the number of monomer types in \mathcal{S} .

A *polymer* is a sequence of monomers $Qm_1m_2 \dots m_nR$ where $m_i \in \Delta$ such that for each pair of adjacent monomers $(w, x, a, b)(c, d, y, z)$, either $\bar{a} = d$ or $\bar{b} = c$. The *length* of a polymer is the number of monomers, including Q and R , it contains. Each pair of adjacent monomer ends $(a, b)(c, d)$ form an *insertion site*. Monomers can be inserted into an insertion site $(a, b)(c, d)$ according to the following rules (see Figure 1):

1. If $\bar{a} = d$, then any monomer $(\bar{b}, e, f, \bar{c})^+$ can be inserted.
2. If $\bar{b} = c$, then any monomer $(e, \bar{a}, \bar{d}, f)^-$ can be inserted.¹

A monomer is inserted after time t , where t is an exponential random variable with rate equal to the concentration of the monomer type. The set of all polymers *constructed* by an insertion system is recursively defined as any polymer constructed by inserting a monomer into a polymer constructed by the system, beginning with the initiator. Note that the insertion rules guarantee by induction that for every insertion site $(a, b)(c, d)$, either $\bar{a} = d$ or $\bar{b} = c$.

¹In [6], this rule is described as a monomer $(\bar{d}, f, e, \bar{a})^-$ that is inserted into the polymer as (e, \bar{a}, \bar{d}, f) .

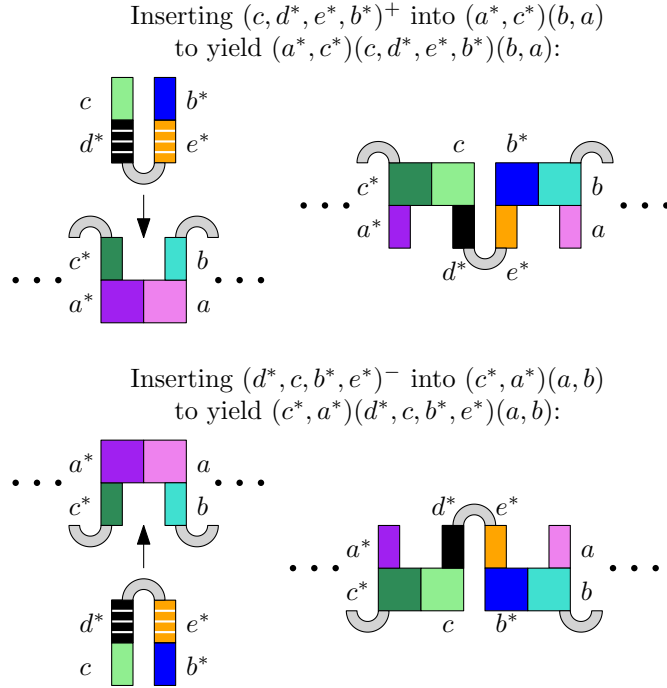


Figure 1: A pictorial interpretation of the two insertion rules for monomers. Loosely based on Figure 2 and corresponding DNA-based implementation of [6].

We say that a polymer is *terminal* if no monomer can be inserted into any insertion site in the polymer, and that an insertion system *deterministically constructs* a polymer P if every polymer constructed by the system is either P or is non-terminal and has length less than that of P (i.e. can become P). The *string representation* of a polymer is the sequence of symbols found on the polymer from left to right, e.g. $(a, b)(b^*, a, d, c)(c^*, a)$ has string representation abb^*adcc^*a . We call the set of string representations of all terminal polymers of an insertion system \mathcal{S} the *language* of \mathcal{S} , denoted $L(\mathcal{S})$.

2.3 Expressive power

Intuitively, a system *expresses* another if the terminal polymers or strings created by the system “look” like the terminal polymers or strings created by the other system. In the simplest instance, an integer-pair grammar \mathcal{G}' is said to *express* a context-free grammar \mathcal{G} if $L(\mathcal{G}') = L(\mathcal{G})$. Similarly, a grammar \mathcal{G} is said to *express* an insertion system \mathcal{S} if $L(\mathcal{S}) = L(\mathcal{G})$, i.e. if the set of string representations of the terminal polymers of \mathcal{S} equals the language of \mathcal{G} .

An insertion system $\mathcal{S} = (\Sigma', \Delta', Q', R')$ is said to express a grammar $\mathcal{G} = (\Sigma, \Gamma, \Delta, S)$ if there exists a function $g : \Sigma' \cup \{s^* : s \in \Sigma'\} \rightarrow \Sigma \cup \{\varepsilon\}$ such that $\{g(s'_1)g(s'_2) \dots g(s'_n) : s'_1 s'_2 \dots s'_n \in L(\mathcal{S})\} = L(\mathcal{G})$. More precisely, we require

that there exists a fixed integer κ such that for any substring $s'_{i+1}s'_{i+2}\dots s'_{i+\kappa}$ in a string in $L(\mathcal{S})$, $\{g(s'_{i+1}), g(s'_{i+2}), \dots, g(s'_{i+\kappa})\} \neq \{\varepsilon\}$. That is, the insertion system symbols mapping to grammar terminal symbols are evenly distributed throughout the polymer. The requirement of a fixed integer κ prevents the possibility of a polymer containing arbitrarily long and irregular regions of “garbage” monomers.

3 The Expressive Power of Insertion Systems

Dabby and Chen proved that any insertion system has a context-free grammar expressing it. They construct such a grammar by creating a non-terminal for every possible insertion site and a production rule for every monomer type insertable into the site. For instance, the insertion site $(a, b)(c^*, a^*)$ and monomer type $(b^*, d^*, e, c)^+$ induce non-terminal symbol $A_{(a,b)(c^*,a^*)}$ and production rule $A_{(a,b)(c^*,a^*)} \rightarrow A_{(a,b)(b^*,d^*)}A_{(e,c)(c^*,a^*)}$. Here we give a reduction in the other direction, resolving in the affirmative the question posed by Dabby and Chen of whether context-free grammars and insertion systems have the same expressive power:

Theorem 3.1. *For every context-free grammar G , there exists an insertion system that expresses G .*

The primary difficulty in proving Theorem 3.1 lies in developing a way to simulate the “complete” replacement that occurs during derivation with the “incomplete” replacement that occurs at an insertion site during insertion. For instance, $bcAbc \Rightarrow bcDDbc$ via a production rule $A \rightarrow DD$ and A is completely replaced by DD . On the other hand, inserting a monomer $(b^*, d, d, c)^+$ into a site $(a, b)(c^*, a^*)$ yields the consecutive sites $(a, b)(b^*, d)$ and $(d, c)(c^*, a^*)$, with $(a, b)(c^*, a^*)$ only partially replaced – the left side of the first site and the right side of second site together form the initial site. This behavior constrains how replacement can be captured by insertion sites, and the κ parameter of the definition of expression (Section 2.3) prevents eliminating the issue via additional insertions.

We overcome this difficulty by proving Theorem 3.1 in two steps. First, we prove that integer-pair grammars, a constrained type of grammar with incomplete replacements, are able to express context-free grammars (Lemma 3.2). Second, we prove integer-pair grammars can be expressed by insertion systems (Lemma 3.3).

Lemma 3.2. *For every context-free grammar \mathcal{G} , there exists an integer-pair grammar that expresses \mathcal{G} .*

Proof. Let $\mathcal{G} = (\Sigma, \Gamma, \Delta, S)$. Let $n = |\Gamma|$. Start by putting \mathcal{G} into Chomsky normal form and then relabeling the non-terminals of \mathcal{G} to A_0, A_1, \dots, A_{n-1} , with $S = A_0$.

Now we define an integer-pair grammar $\mathcal{G}' = (\Sigma', \Gamma', \Delta', S')$ such that $L(\mathcal{G}') = L(\mathcal{G})$. Let $\Sigma' = \Sigma$ and $\Gamma' = \{(a, d) : 0 \leq a, d < n\}$. For each production rule

$A_i \rightarrow A_j A_k$ in Δ , add to Δ' the set of rules $(a, d) \rightarrow (a, b)(c, d)$, with $0 \leq a < n$, $d = (i - a) \bmod n$, $b = (j - a) \bmod n$, and $c = (k - d) \bmod n$. For each production rule $A_i \rightarrow t$ in Δ , add to Δ' the set of rules $(a, d) \rightarrow t$, with $0 \leq a < n$ and $d = (i - a) \bmod n$. Let $S' = (0, 0)$.

We claim that a partial derivation P' of \mathcal{G}' exists if and only if the partial derivation P obtained by replacing each non-terminal (a, d) in P' with $A_{(a+d) \bmod n}$ is a partial derivation of \mathcal{G} . By construction, a rule $(a, d) \rightarrow (a, b)(c, d)$ is in Δ' if and only if the rule $A_{(a+d) \bmod n} \rightarrow A_{(a+b) \bmod n} A_{(c+d) \bmod n}$ is in Δ . Similarly, a rule $(a, d) \rightarrow t$ is in Δ' if and only if the rule $A_{(a+d) \bmod n} \rightarrow t$ is in Δ . Also, $S' = (0, 0)$ and $S = A_{(0+0) \bmod n}$. So the claim holds by induction.

Since the set of all partial derivations of P' are equal to those of P , the completed derivations are as well and $L(S') = L(S)$. So \mathcal{G}' expresses \mathcal{G} . \square \square

Lemma 3.3. *For every integer-pair grammar \mathcal{G} , there exists an insertion system that expresses \mathcal{G} .*

Proof. Let $\mathcal{G} = (\Sigma, \Gamma, \Delta, S)$. The integer-pair grammar \mathcal{G} is expressed by an insertion system $\mathcal{S} = (\Sigma', \Delta', Q', R')$ that we now define. Let $\Sigma' = \{s_a, s_b : (a, b) \in \Gamma\} \cup \{u, x\} \cup \Sigma$. Let $\Delta' = \Delta'_1 \cup \Delta'_2 \cup \Delta'_3 \cup \Delta'_4$, where

$$\Delta'_1 = \{(s_b, u, s_b^*, x)^- : (a, d) \rightarrow (a, b)(c, d) \in \Delta\}$$

$$\Delta'_2 = \{(s_a, s_b, s_c^*, s_d^*)^+ : (a, d) \rightarrow (a, b)(c, d) \in \Delta\}$$

$$\Delta'_3 = \{(x, s_c, u^*, s_c^*)^- : (a, d) \rightarrow (a, b)(c, d) \in \Delta\}$$

$$\Delta'_4 = \{(s_a, t, x, s_d^*)^+ : (a, d) \rightarrow t \in \Delta\}$$

We give each monomer type equal concentration, although the precise concentrations are not important for expressive power. Let $Q' = (u^*, a^*)$ and $R' = (b, u)$, where $S = (a, b)$.

Insertion types. We start by proving that for any polymer constructed by \mathcal{S} , only the following types of insertions of a monomer m_2 between two adjacent monomers $m_1 m_3$ are possible:

1. $m_1 \in \Delta'_2, m_2 \in \Delta'_3, m_3 \in \Delta'_1$
2. $m_1 \in \Delta'_3, m_2 \in \Delta'_2 \cup \Delta'_4, m_3 \in \Delta'_1$
3. $m_1 \in \Delta'_3, m_2 \in \Delta'_1, m_3 \in \Delta'_2$

Moreover, we claim that for every adjacent $m_1 m_3$ pair satisfying one of these conditions, an insertion *is* possible. That is, there is a monomer m_2 that can be inserted, necessarily from the monomer subset specified.

Consider each possible combination of $m_1 \in \Delta'_i$ and $m_3 \in \Delta'_j$, respectively, with $i, j \in \{1, 2, 3, 4\}$. Observe that for an insertion to occur at insertion site $(a, b)(c, d)$, the symbols \bar{a} , \bar{b} , \bar{c} , and \bar{d} must each occur on some monomer. Then since x^* and t^* do not appear on any monomers, any i, j with $i \in \{1, 4\}$ or

$j \in \{3, 4\}$ cannot occur. This leaves monomer pairs (Δ'_i, Δ'_j) with $(i, j) \in \{(2, 1), (2, 2), (3, 1), (3, 2)\}$.

Insertion sites between (Δ'_2, Δ'_1) pairs have the form $(s_c^*, s_d^*)(s_b, u)$, so an inserted monomer must have the form $(s_e, s_c, s_u^*, s_f)^-$ and is in Δ'_3 . An insertion site $(s_c^*, s_d^*)(s_b, u)$ implies a rule of the form $(e, d) \rightarrow (e, f)(c, d)$ in Δ , so there exists a monomer $(x, s_c, u^*, s_c^*)^- \in \Delta'_3$ that can be inserted.

Insertion sites between (Δ'_3, Δ'_2) pairs have the form $(u^*, s_c^*)(s_a, s_b)$, so an inserted monomer must have the form $(_, u, s_b^*, _)^-$ and thus is in Δ'_1 . An insertion site $(u^*, s_c^*)(s_a, s_b)$ implies a rule of the form $(a, d) \rightarrow (a, b)(e, d)$ in Γ , so there exists a monomer $(s_b, u, s_b^*, x)^- \in \Delta'_1$ that can be inserted.

Insertion sites between (Δ'_2, Δ'_2) pairs can only occur once a monomer $m_2 \in \Delta'_2$ has been inserted between a pair of adjacent monomers $m_1 m_3$ with either $m_1 \in \Delta'_2$ or $m_3 \in \Delta'_2$, but not both. But we just proved that all such possible insertions only permit $m_2 \in \Delta'_3 \cup \Delta'_1$. Moreover, the initial insertion site between Q' and R' has the form $(u^*, s_a^*)(s_b, u)$ of an insertion site with $m_1 \in \Delta'_3$ and $m_3 \in \Delta'_1$. So no pair of adjacent monomers $m_1 m_3$ are ever both from Δ'_2 and no insertion site between (Δ'_2, Δ'_2) pairs can ever exist.

Insertion sites between (Δ'_3, Δ'_1) pairs have the form $(u^*, s_c^*)(s_b, u)$, so an inserted monomer must have the form $(s_c, _, _, b^*)^+$ or $(_, u, u^*, _)^-$ and is in Δ'_2 or Δ'_4 . We show by induction that for each such insertion site $(u^*, s_c^*)(s_b, u)$ that $(c, b) \in \Gamma$. First, observe that this is true for the insertion site $(u^*, s_a^*)(s_b, u)$ between Q' and R' , since $(a, b) = S \in \Gamma$. Next, suppose this is true for all insertion sites of some polymer and a monomer $m_2 \in \Delta'_2 \cup \Delta'_4$ is about to be inserted into the polymer between monomers from Δ'_3 and Δ'_1 . Inserting a monomer $m_2 \in \Delta'_4$ only reduces the set of insertion sites between monomers in Δ'_3 and Δ'_1 , and the inductive hypothesis holds. Inserting a monomer $m_2 \in \Delta'_2$ induces new (Δ'_3, Δ'_2) and (Δ'_2, Δ'_1) insertion site pairs between $m_1 m_2$ and $m_2 m_3$. These pairs must accept two monomers $m_4 \in \Delta_1$ and $m_5 \in \Delta_3$, inducing a sequence of monomers $m_1 m_4 m_2 m_5 m_3$ with adjacent pairs (Δ'_3, Δ'_1) , (Δ'_1, Δ'_2) , (Δ'_2, Δ'_3) , (Δ'_3, Δ'_1) . Only the first and last pairs permit insertion and both are (Δ'_3, Δ'_1) pairs.

Now consider the details of the three insertions yielding $m_1 m_4 m_2 m_5 m_3$, starting with $m_1 m_3$. The initial insertion site $m_1 m_3$ must have the form $(u^*, s_a^*)(s_d, u)$. So the sequence of insertions has the following form, with the last two insertions interchangeable. The symbol \diamond is used to indicate the site being modified and the inserted monomer shown in bold:

$$\begin{aligned}
& (u^*, s_a^*) \diamond (s_d, u) \\
& (u^*, s_a^*) \diamond (\mathbf{s_a}, \mathbf{s_b}, \mathbf{s_c^*}, \mathbf{s_d^*})(s_d, u) \\
& (u^*, s_a^*)(\mathbf{s_b}, \mathbf{u}, \mathbf{s_b^*}, \mathbf{x})(s_a, s_b, s_c^*, s_d^*) \diamond (s_d, u) \\
& (u^*, s_a^*) \diamond (s_b, u, s_b^*, x)(s_a, s_b, s_c^*, s_d^*)(\mathbf{x}, \mathbf{s_c}, \mathbf{u^*}, \mathbf{s_c^*}) \diamond (s_d, u)
\end{aligned}$$

Notice the two resulting (Δ'_3, Δ'_1) pair insertion sites $(u^*, s_a^*)(s_b, u)$ and $(u^*, s_c^*)(s_d, u)$. Assume, by induction, that the monomer m_2 must exist. So

there is a rule $(a, d) \rightarrow (a, b)(c, d) \in \Delta$ and $(a, b), (c, d) \in \Gamma$, fulfilling the inductive hypothesis. So for every insertion site $(u^*, s_c^*)(s_b, u)$ between a (Δ'_3, Δ'_1) pair there exists a non-terminal $(c, b) \in \Gamma$. So for every adjacent monomer pair $m_1 m_3$ with $m_1 \in \Delta'_3$ and $m_3 \in \Delta'_1$, there exists a monomer $m_2 \in \Delta'_2 \cup \Delta'_4$ that can be inserted between m_1 and m_3 .

Partial derivations and terminal polymers. Next, consider the sequence of insertion sites between (Δ'_3, Δ'_1) pairs in a polymer constructed by a modified version of \mathcal{S} lacking the monomers of Δ'_4 . We claim that a polymer with a sequence $(u^*, s_{a_1}^*)(s_{b_1}, u), (u^*, s_{a_2}^*)(s_{b_2}, u), \dots, (u^*, s_{a_i}^*)(s_{b_i}, u)$ of (Δ'_3, Δ'_1) insertion sites is constructed if and only if there is a partial derivation $(a_1, b_1)(a_2, b_2) \dots (a_i, b_i)$ of a string in $L(\mathcal{G})$. This follows directly from the previous proof by observing that two new adjacent (Δ'_3, Δ'_1) pair insertion sites $(u^*, s_a^*)(s_b, u)$ and $(u^*, s_c^*)(s_d, u)$ can replace a (Δ'_3, Δ'_1) pair insertion site if and only if there exists a rule $(a, d) \rightarrow (a, b)(c, d) \in \Delta$.

Observe that any string in $L(\mathcal{G})$ can be derived by first deriving a partial derivation containing only non-terminals, then applying only rules of the form $(a, d) \rightarrow t$. Similarly, since the monomers of Δ'_4 never form half of a valid insertion site, any terminal polymer of \mathcal{S} can be constructed by first generating a polymer containing only monomers in $\Delta'_1 \cup \Delta'_2 \cup \Delta'_3$, then only inserting monomers from Δ'_4 . Also note that the types of insertions possible in \mathcal{S} imply that in any terminal polymer, any triple of adjacent monomers $m_1 m_2 m_3$ with $m_1 \in \Delta'_i, m_2 \in \Delta'_j$, and $m_3 \in \Delta'_k$, that $(i, j, k) \in \{(4, 1, 2), (1, 2, 3), (2, 3, 4), (3, 4, 1)\}$, with the first and last monomers of the polymer in Δ'_4 .

Expression. Define the following piecewise function $g : \Sigma' \cup \{s^* : s \in \Sigma'\} \rightarrow \Sigma \cup \{\varepsilon\}$ that maps to ε except for second symbols of monomers in Δ'_4 .

$$g(s) = \begin{cases} t, & \text{if } t \in \Sigma \\ \varepsilon, & \text{otherwise} \end{cases}$$

Observe that every string in $L(\mathcal{S})$ has length $2 + 4 \cdot (4n - 3) + 2 = 16n - 8$ for some $n \geq 0$. Also, for each string $s'_1 s'_2 \dots s'_{16n-8} \in L(\mathcal{S})$, $g(s'_1)g(s'_2) \dots g(s'_{16n-8}) = \varepsilon^3 t_1 \varepsilon^{16} t_2 \varepsilon^{16} \dots t_n \varepsilon^5$. There is a terminal polymer with string representation in $L(\mathcal{S})$ yielding the sequence $s_1 s_2 \dots s_n$ if and only if the polymer can be constructed by first generating a terminal polymer excluding Δ'_4 monomers with a sequence of (Δ'_3, Δ'_1) insertion pairs $(a_1, b_1)(a_2, b_2) \dots (a_n, b_n)$ followed by a sequence of insertions of monomers from Δ'_4 with second symbols $t_1 t_2 \dots t_n$. Such a generation is possible if and only if $(a_1, b_1)(a_2, b_2) \dots (a_n, b_n)$ is a partial derivation of a string in $L(\mathcal{G})$ and $(a_1, b_1) \rightarrow t_1, (a_2, b_2) \rightarrow t_2, \dots, (a_n, b_n) \rightarrow t_n \in \Delta$. So applying the function g to the string representations of the terminal polymers of \mathcal{S} gives $L(\mathcal{G})$, i.e. $L(\mathcal{S}) = L(\mathcal{G})$. Moreover, the second symbol in every fourth monomer in a terminal polymer of \mathcal{S} maps to a symbol of Σ using g . So \mathcal{S} expresses \mathcal{G} with the function g and $\kappa = 16$. \square \square

4 Positive Results for Polymer Growth

Dabby and Chen also consider the size and speed of constructing finite polymers. They give a construction achieving the following result:

Theorem 4.1 ([6]). *For any positive integer r , there exists an insertion system with $O(r^2)$ monomer types that deterministically constructs a polymer of length $n = 2^{\Theta(r)}$ in $O(\log^3 n)$ expected time. Moreover, the expected time has an exponentially decaying tail probability.*

We begin this section by improving on this construction significantly in both polymer length and expected running time (Theorem 4.2). In Section 5 we prove that our construction is the best possible with respect to both the polymer length and construction time for deterministic systems.

Theorem 4.2. *For any positive integer r , there exists an insertion system with $O(r^2)$ monomer types that deterministically constructs a polymer of length $n = 2^{\Theta(r^3)}$ in $O(\log^{5/3}(n))$ expected time. Moreover, the expected time has an exponentially decaying tail probability.*

Proof. The approach is to implement a three variable counter where each variable ranges over the values 0 to r , effectively carrying out the execution of a triple for-loop. Insertion sites of the form $(s_a, s_b)(s_c, s_a^*)$ are used to encode the state of the counter, where a , b , and c are the variables of the outer, inner, and middle loops, respectively. Three types of variable increments are carried out by the counter:

Inner: If $b < r$, then $(s_a, s_b)(s_c, s_a^*) \rightsquigarrow (s_a, s_{b+1})(s_c, s_a^*)$.

Middle: If $b = r$ and $c < r$, then $(s_a, s_b)(s_c, s_a^*) \rightsquigarrow (s_a, s_0)(s_{c+1}, s_a^*)$.

Outer: If $b = c = r$ and $a < r$, then $(s_a, s_b)(s_c, s_a^*) \rightsquigarrow (s_{a+1}, s_0)(s_0, s_{a+1}^*)$.

A site is *modified* by a sequence of monomer insertions that yields a new usable site where all other sites created by the insertion sequence are unusable. For instance, we modify a site $(s_a, \mathbf{s}_b)(s_c, s_a^*)$ to become $(s_a, \mathbf{s}_d)(s_c, s_a^*)$, written $(s_a, s_b)(s_c, s_a^*) \rightsquigarrow (s_a, s_d)(s_c, s_a^*)$, by adding the monomer types $(s_b^*, x, u, s_c^*)^+$ and $(x, u^*, s_a, s_d)^-$ to the system, where x is a special symbol whose complement is not found on any monomer. These two monomer types cause the following sequence of insertions, using \diamond to indicate the site being modified and the inserted monomer shown in bold:

$$\begin{aligned} & (s_a, s_b) \diamond (s_c, s_a^*) \\ & (s_a, s_b)(\mathbf{s}_b^*, \mathbf{x}, \mathbf{u}, \mathbf{s}_c^*) \diamond (s_c, s_a^*) \\ & (s_a, s_b)(s_b^*, x, u, s_c^*)(\mathbf{x}, \mathbf{u}^*, \mathbf{s}_a, \mathbf{s}_d) \diamond (s_c, s_a^*) \end{aligned}$$

Replacement	Monomers
$(s_a, \mathbf{s}_b)(s_c, s_a^*) \rightsquigarrow (s_a, \mathbf{s}_d)(s_c, s_a^*)$	$(s_b^*, x, u, s_c^*)^+, (x, u^*, s_a, s_d)^-$
$(s_a, s_b)(\mathbf{s}_c, s_a^*) \rightsquigarrow (s_a, s_b)(\mathbf{s}_d, s_a^*)$	$(s_b^*, u, x, s_c^*)^+, (s_d, s_a^*, u^*, x)^-$
$(\mathbf{s}_b, s_a)(s_a^*, s_c) \rightsquigarrow (\mathbf{s}_d, s_a)(s_a^*, s_c)$	$(x, s_b^*, s_c^*, u)^-, (u^*, x, s_d, s_a)^+$
$(s_b, s_a)(s_a^*, \mathbf{s}_c) \rightsquigarrow (s_b, s_a)(s_a^*, \mathbf{s}_d)$	$(u, s_b^*, s_c^*, x)^-, (s_a^*, s_d, x, u^*)^+$

Table 1: The four types of replacement steps and monomer pairs that implement them. The symbol u can be any symbol, and x is a special symbol whose complement does not appear on any monomer.

We call this simple modification, where a single symbol in the insertion site is replaced with another symbol, a *replacement*. There are four types of replacements, seen in Table 1, that can each be implemented by a pair of corresponding monomers.

Each of the three increment types are implemented using a sequence of site modifications. The resulting triple for-loop carries out a sequence of $\Theta(r^3)$ insertions, constructing a $\Theta(r^3)$ -length polymer. A $2^{\Theta(r^3)}$ -length polymer is achieved by simultaneously duplicating each site during each inner increment. Because the for-loop runs for $\Theta(r^3)$ steps and duplicates at a constant fraction of these steps (those with $0 \leq b < r$), the number of counters reaching the final $a = b = c = r$ state is $2^{\Theta(r^3)}$. In the remainder of the proof, we detail the implementation of each increment type, starting with the simplest: middle increments.

Middle increment. A middle increment of a site $(s_a, s_b)(s_c, s_a^*)$ occurs when the site has the form $(s_a, s_r)(s_c, s_a^*)$ with $0 \leq c < r$, performing the modification $(s_a, s_r)(s_c, s_a^*) \rightsquigarrow (s_a, s_0)(s_{c+1}, s_a^*)$. We implement middle increments using a sequence of three replacements:

$$(s_a, s_r)(s_c, s_a^*) \xrightarrow{1} (s_a, s_r)(s_{f_1(c)}, s_a^*) \xrightarrow{2} (s_a, s_0)(s_{f_1(c)}, s_a^*) \xrightarrow{3} (s_a, s_0)(s_{c+1}, s_a^*)$$

where $f_i(n) = n + 2ir^2$. The use of f is to avoid unintended interactions between monomers, since for any n_1, n_2 with $0 \leq n_1, n_2 \leq r$, $f_i(n_1) \neq f_j(n_2)$ for all $i \neq j$. Compiling this sequence of replacements into monomer types gives the following monomers:

Step 1: $(s_r^*, s_{f_2(c)}, x, s_c^*)^+$ and $(s_{f_1(c)}, s_a^*, s_{f_2(c)}, x)^-$.

Step 2: $(s_r^*, x, s_{f_3(c)}, s_{f_1(c)}^*)^+$ and $(x, s_{f_3(c)}^*, s_a, s_0)^-$.

Step 3: $(s_0^*, s_{f_4(c+1)}, x, s_{f_1(c)}^*)^+$ and $(s_{c+1}, s_a^*, s_{f_4(c+1)}, x)^-$.

This set of monomers results in the following sequence of six insertions:

$$(s_a, s_r) \diamond (s_c, s_a^*)$$

$$\begin{aligned}
& (s_a, s_r) \diamond (\mathbf{s}_r^*, \mathbf{s}_{f_2(c)}, \mathbf{x}, \mathbf{s}_c^*) (s_c, s_a^*) \\
& (s_a, s_r) \diamond (\mathbf{s}_{f_1(c)}, \mathbf{s}_a^*, \mathbf{s}_{f_2(c)}^*, \mathbf{x}) (s_r^*, s_{f_2(c)}, x, s_c^*) (s_c, s_a^*) \\
& (s_a, s_r) \diamond (s_{f_1(c)}, s_a^*) \\
& (s_a, s_r) (\mathbf{s}_r^*, \mathbf{x}, \mathbf{s}_{f_3(c)}, \mathbf{s}_{f_1(c)}^*) \diamond (s_{f_1(c)}, s_a^*) \\
& (s_a, s_r) (s_r^*, x, s_{f_3(c)}, s_{f_1(c)}^*) (\mathbf{x}, \mathbf{s}_{f_3(c)}^*, \mathbf{s}_a, \mathbf{s}_0) \diamond (s_{f_1(c)}, s_a^*) \\
& (s_a, s_0) \diamond (s_{f_1(c)}, s_a^*) \\
& (s_a, s_0) \diamond (\mathbf{s}_0^*, \mathbf{s}_{f_4(c+1)}, \mathbf{x}, \mathbf{s}_{f_1(c)}^*) (s_{f_1(c)}, s_a^*) \\
& (s_a, s_0) \diamond (\mathbf{s}_{c+1}, \mathbf{s}_a^*, \mathbf{s}_{f_4(c+1)}^*, \mathbf{x}) (s_0^*, s_{f_4(c+1)}, x, s_{f_1(c)}^*) (s_{f_1(c)}, s_a^*) \\
& (s_a, s_0) \diamond (s_{c+1}, s_a^*)
\end{aligned}$$

Since each inserted monomer has an instance of x , all other insertion sites created are unusable. This is true of the insertions used for outer increments and duplications as well.

Outer increment. An outer increment of the site $(s_a, s_b)(s_c, s_a^*)$ occurs when the site has the form $(s_a, s_r)(s_r, s_a^*)$ with $0 \leq a < r$. We implement this step using a two-phase sequence of three normal replacements and a special quadruple replacement (Step 3):

$$\begin{aligned}
& (s_a, s_r)(s_r, s_a^*) \xrightarrow{1} (s_a, s_{f_5(a)})(s_r, s_a^*) \xrightarrow{2} (s_a, s_{f_5(a)})(s_{f_5(a)}^*, s_a^*) \\
& (s_a, s_{f_5(a)})(s_{f_5(a)}^*, s_a^*) \xrightarrow{3} (s_{a+1}, s_{f_5(0)})(s_0, s_{a+1}^*) \xrightarrow{4} (s_{a+1}, s_0)(s_0, s_{a+1}^*)
\end{aligned}$$

At each step, a (necessary) complementary pair of symbols is maintained, which results in a sequence of more than 4 replacements. As with inner and middle increments, we compile replacement steps 1, 2, and 4 into monomers using Table 1. Step 3 is a special pair of monomers.

Step 1: $(s_r^*, x, s_{f_6(r)}, s_r^*)^+$ and $(x, s_{f_6(r)}^*, s_a, s_{f_5(a)})^-$.

Step 2: $(s_{f_5(a)}^*, s_{f_7(r)}^*, x, s_r^*)^+$ and $(s_{f_5(a)}^*, s_a^*, s_{f_7(r)}, x)^-$.

Step 3: $(s_{f_5(a)}^*, x, s_{a+1}, s_{f_5(a)})^+$ and $(s_0, s_{a+1}^*, s_a, x)^-$.

Step 4: $(s_{f_5(a)}^*, x, s_{f_7(r)}, s_0^*)^+$ and $(x, s_{f_7(r)}^*, s_{a+1}, s_0)^-$.

Here is the sequence of insertions, using \diamond to indicate the site being modified and the inserted monomer shown in bold:

$$\begin{aligned}
& (s_a, s_r) \diamond (s_r, s_a^*) \\
& (s_a, s_r) (\mathbf{s}_r^*, \mathbf{x}, \mathbf{s}_{f_6(r)}, \mathbf{s}_r^*) \diamond (s_r, s_a^*) \\
& (s_a, s_r) (s_r^*, x, s_{f_6(r)}, s_r^*) (\mathbf{x}, \mathbf{s}_{f_6(r)}^*, \mathbf{s}_a, \mathbf{s}_{f_5(a)}) \diamond (s_r, s_a^*) \\
& (s_a, s_{f_5(a)}) \diamond (s_r, s_a^*)
\end{aligned}$$

$$\begin{aligned}
& (s_a, s_{f_5(a)}) \diamond (s_{f_5(a)}^*, s_{f_7(r)}^*, x, s_r^*)(s_r, s_a^*) \\
& (s_a, s_{f_5(a)}) \diamond (s_{f_5(a)}^*, s_a^*, s_{f_7(r)}^*, x)(s_{f_5(a)}^*, s_{f_7(r)}^*, x, s_r^*)(s_r, s_a^*) \\
& (s_a, s_{f_5(a)}) \diamond (s_{f_5(a)}^*, s_a^*) \\
& (s_a, s_{f_5(a)})(s_{f_5(a)}^*, x, s_{a+1}, s_{f_5(a)}) \diamond (s_{f_5(a)}^*, s_a^*) \\
& (s_a, s_{f_5(a)})(s_{f_5(a)}^*, x, s_{a+1}, s_{f_5(a)}) \diamond (s_0, s_{a+1}^*, s_a, x)(s_{f_5(a)}^*, s_a^*) \\
& (s_{a+1}, s_{f_5(a)}) \diamond (s_0, s_{a+1}^*) \\
& (s_{a+1}, s_{f_5(a)})(s_{f_5(a)}^*, x, s_{f_7(r)}^*, s_0^*) \diamond (s_0, s_{a+1}^*) \\
& (s_{a+1}, s_{f_5(a)})(s_{f_5(a)}^*, x, s_{f_7(r)}^*, s_0^*)(x, s_{f_7(r)}^*, s_{a+1}, s_0) \diamond (s_0, s_{a+1}^*) \\
& (s_{a+1}, s_0) \diamond (s_0, s_{a+1}^*)
\end{aligned}$$

Inner increment. The inner increment has two phases. The first phase (Steps 1-2) performs duplication, modifying the initial site to a pair of sites: $(s_a, s_b)(s_c, s_a^*) \rightsquigarrow (s_a, s_b)(s_{f_8(c)}, s_a^*) \dots (s_a, s_{b+1})(s_c, s_a^*)$, yielding an incremented version of the original site and one other site. The second phase (Steps 3-5) is $(s_a, s_b)(s_{f_8(c)}, s_a^*) \rightsquigarrow (s_a, s_{b+1})(s_c, s_a^*)$, transforming the second site into an incremented version of the original site.

For the first phase, we use the three monomers:

Step 1: $(s_b^*, s_{f_8(c)}, s_{f_8(b+1)}, s_c^*)^+$.

Step 2: $(s_{f_8(c)}, s_a^*, s_{f_8(c)}^*, x)^-$ and $(x, s_{f_8(b+1)}^*, s_a, s_{b+1})^-$.

Here is the sequence of insertions:

$$\begin{aligned}
& (s_a, s_b) \diamond (s_c, s_a^*) \\
& (s_a, s_b) \diamond (s_b^*, s_{f_8(c)}, s_{f_8(b+1)}, s_c^*) \diamond (s_c, s_a^*) \\
& (s_a, s_b) \diamond (s_{f_8(c)}, s_a^*, s_{f_8(c)}^*, x)(s_b^*, s_{f_8(c)}, s_{f_8(b+1)}, s_c^*) \diamond (s_c, s_a^*) \\
& (s_a, s_b) \diamond (s_{f_8(c)}, s_a^*, s_{f_8(c)}^*, x)(s_b^*, s_{f_8(c)}, s_{f_8(b+1)}, s_c^*)(x, s_{f_8(b+1)}^*, s_a, s_{b+1}) \diamond (s_c, s_a^*) \\
& (s_a, s_b) \diamond (s_{f_8(c)}, s_a^*) \dots (s_a, s_{b+1}) \diamond (s_c, s_a^*)
\end{aligned}$$

The last two insertions occur independently and may happen in the opposite order of the sequence depicted here. In the second phase, the site $(s_a, s_b)(s_{f_8(c)}, s_a^*)$ is transformed into $(s_a, s_{b+1})(s_c, s_a^*)$ by a sequence of replacement steps:

$$(s_a, s_b)(s_{f_8(c)}, s_a^*) \xrightarrow{3} (s_a, s_{f_9(b)})(s_{f_8(c)}, s_a^*) \xrightarrow{4} (s_a, s_{f_9(b)})(s_c, s_a^*) \xrightarrow{5} (s_a, s_{b+1})(s_c, s_a^*)$$

As with previous sequences of replacement steps, we compile this sequence into a set of monomers:

Step 3: $(s_b^*, x, s_{f_{10}(b)}, s_{f_8(c)}^*)^+$ and $(x, s_{f_{10}(b)}^*, s_a, s_{f_9(b)})^-$.

Step 4: $(s_{f_9(b)}^*, s_{f_{11}(c)}, x, s_{f_8(c)}^*)^+$ and $(s_c, s_a^*, s_{f_{11}(c)}^*, x)^-$.

Step 5: $(s_{f_9(b)}^*, x, s_{f_{12}(b+1)}, s_c^*)^+$ and $(x, s_{f_{12}(b+1)}^*, s_a, s_{b+1})^-$.

Here is the resulting sequence of insertions:

$$\begin{aligned}
& (s_a, s_b) \diamond (s_{f_8(c)}, s_a^*) \\
& (s_a, s_b)(s_b^*, x, s_{f_{10}(b)}, s_{f_8(c)}^*) \diamond (s_{f_8(c)}, s_a^*) \\
& (s_a, s_b)(s_b^*, x, s_{f_{10}(b)}, s_{f_8(c)}^*)(x, s_{f_{10}(b)}^*, s_a, s_{f_9(b)}) \diamond (s_{f_8(c)}, s_a^*) \\
& (s_a, s_{f_9(b)}) \diamond (s_{f_8(c)}, s_a^*) \\
& (s_a, s_{f_9(b)}) \diamond (s_{f_9(b)}^*, s_{f_{11}(c)}, x, s_{f_8(c)}^*)(s_{f_8(c)}, s_a^*) \\
& (s_a, s_{f_9(b)}) \diamond (s_c, s_a^*, s_{f_{11}(c)}^*, x)(s_{f_9(b)}^*, s_{f_{11}(c)}, x, s_{f_8(c)}^*)(s_{f_8(c)}, s_a^*) \\
& (s_a, s_{f_9(b)}) \diamond (s_c, s_a^*) \\
& (s_a, s_{f_9(b)})(s_{f_9(b)}^*, x, s_{f_{12}(b+1)}, s_c^*) \diamond (s_c, s_a^*) \\
& (s_a, s_{f_9(b)})(s_{f_9(b)}^*, x, s_{f_{12}(b+1)}, s_c^*)(x, s_{f_{12}(b+1)}^*, s_a, s_{b+1}) \diamond (s_c, s_a^*) \\
& (s_a, s_{b+1}) \diamond (s_c, s_a^*)
\end{aligned}$$

When combined, the two phases of duplication modify $(s_a, s_b)(s_c, s_a^*)$ to become $(s_a, s_{b+1})(s_c, s_a^*) \dots (s_a, s_{b+1})(s_c, s_a^*)$, where all sites between the duplicated sites are unusable. Notice that although we need to duplicate $\Theta(r^3)$ distinct sites, only $\Theta(r^2)$ monomers are used in the implementation since each monomer either does not depend on a , e.g. $(s_b^*, x, s_{f_{10}(b)}, s_{f_8(c)}^*)^+$, or does not depend on c , e.g. $(x, s_{f_{10}(b)}^*, s_a, s_{f_9(b)})^-$.

Putting it together. The system starts with the initiator $(s_0, s_0)(s_0, s_0^*)$. Each increment of the counter occurs either through a middle increment, outer increment, or a duplication. The total set of monomers is seen in Table 2. There are at most $(r+1)^2$ monomer types in each family (each row of Table 2) and $O(r^2)$ monomer types total.

The system is deterministic if no pair of monomers can be inserted into any insertion site appearing during construction. It can be verified by an inspection of Table 2 that any pair of positive monomers have a distinct pair of first and fourth symbols, and any pair of negative monomers have a distinct pair of second and third symbols. So all insertion sites with only one pair of complementary symbols have at most one insertable monomer, and moreover create only those sites listed via the insertion sequences.

The first insertion site with two pairs of complementary symbols must be one created by a sequence of insertions into sites with a single pair of complementary symbols (that have only one insertable monomer). So this site has the form $(s_a, s_{f_6(a)})(s_{f_6(a)}^*, s_a^*)$ for some $0 \leq a < r$, created during the outer increment step. For each such site, a positive monomer $(s_{f_6(a)}^*, x, s_{a+1}, s_{f_6(a)})^+$ can attach, but no negative monomer has second and third symbols s_a^* and s_a . So these sites have only one insertable monomer and are the only insertion sites with two pairs of complementary symbols. So all sites with two pairs of complementary

Step	Inner monomer types ($b < r$)	
1	$(s_b^*, s_{f_8(c)}, s_{f_8(b+1)}, s_c^*)^+$	
2	$(s_{f_8(c)}, s_a^*, s_{f_8(c)}, x)^-$	$(x, s_{f_8(b+1)}^*, s_a, s_{b+1})^-$
3	$(s_b^*, x, s_{f_{10}(b)}, s_{f_8(c)}^*)^+$	$(x, s_{f_{10}(b)}^*, s_a, s_{f_9(b)})^-$
4	$(s_{f_9(b)}^*, s_{f_{11}(c)}, x, s_{f_8(c)}^*)^+$	$(s_c, s_a^*, s_{f_{11}(c)}^*, x)^-$
5	$(s_{f_9(b)}^*, x, s_{f_{12}(b+1)}, s_c^*)^+$	$(x, s_{f_{12}(b+1)}^*, s_a, s_{b+1})^-$
Step	Middle monomer types ($c < r$)	
1	$(s_r^*, s_{f_2(c)}, x, s_c^*)^+$	$(s_{f_1(c)}, s_a^*, s_{f_2(c)}^*, x)^-$
2	$(s_r^*, x, s_{f_3(c)}, s_{f_1(c)}^*)^+$	$(x, s_{f_3(c)}^*, s_a, s_0)^-$
3	$(s_0^*, s_{f_4(c+1)}, x, s_{f_1(c)}^*)^+$	$(s_{c+1}, s_a^*, s_{f_4(c+1)}^*, x)^-$
Step	Outer monomer types ($a < r$)	
1	$(s_r^*, x, s_{f_6(r)}, s_r^*)^+$	$(x, s_{f_6(r)}^*, s_a, s_{f_5(a)})^-$
2	$(s_{f_5(a)}^*, s_{f_7(r)}^*, x, s_r^*)^+$	$(s_{f_5(a)}^*, s_a^*, s_{f_7(r)}, x)^-$
3	$(s_{f_5(a)}^*, x, s_{a+1}, s_{f_5(a)})^+$	$(s_0, s_{a+1}^*, s_a, x)^-$
4	$(s_{f_5(a)}^*, x, s_{f_7(r)}, s_0^*)^+$	$(x, s_{f_7(r)}^*, s_{a+1}, s_0)^-$

Table 2: The set of all monomer types used to deterministically construct a monomer of size $2^{\Theta(r^3)}$ using $O(r^2)$ monomer types.

symbols also have at most one insertable monomer. Then since all sites occurring in the system have at most one insertable monomer, the system is deterministic.

The size P_i of a subpolymer with an initiator encoding some value i between 0 and $(r+1)^3 - 1$ can be bounded by $2P_{i+2} + 9 \leq P_i \leq 2P_{i+1} + 9$, since either $i+1$ or $i+2$ is an inner increment step and no step inserts more than 9 monomers. Moreover, $P_{(r+1)^3-2} \geq 1$. So $P_0 + 2$, the size of the terminal polymer, is $2^{\Theta(r^3)}$.

Running time. Define the concentration of each monomer type to be equal. There are $12r^2 + 24r + 3 \leq 39r^2$ monomer types, so each monomer type has concentration at least $1/(39r^2)$. The polymer is complete as soon as every counter's variables have reached the value $a = b = c = r$, i.e. every site encoding a counter has been modified to become $(s_r, s_r)(s_r, s_r^*)$ and the monomer $(s_r^*, x, s_{f_r(r)}, s_r^*)^+$ has been inserted.

There are fewer than 2^{r^3} such insertions, and each insertion requires at most $9 \cdot (r+1)^3 \leq 72r^3$ previous insertions to occur. So an upper bound on the expected time T_r for each such insertion is described as a sum of $72r^3$ random variables, each with expected time $39r^2$. The Chernoff bound for exponential random variables implies the following upper bound on T_r :

$$\begin{aligned} \text{Prob}[T_r > 39r^2 \cdot 72r^3(1 + \delta)] &\leq e^{-39 \cdot 72r^5 \delta^2 / (2 + \delta)} \\ &\leq e^{-r^5 \delta^2 / (2 + \delta)} \\ &\leq e^{-r^5 \delta^2 / (2\delta)} \text{ for all } \delta \geq 2 \\ &\leq e^{-r^5 \delta / 2} \end{aligned}$$

Let T_{S_r} be the total running time of the system. Then we can bound T_{S_r} from above using the bound for T_r :

$$\begin{aligned} \text{Prob}[T_{S_r} > 39r^2 \cdot 72r^3(1 + \delta)] &\leq 2^{r^3} \cdot e^{-r^5 \delta / 2} \\ &\leq 2^{r^3} 2^{-r^5 \delta / 2} \\ &\leq 2^{r^3 - r^5 \delta / 2} \\ &\leq 2^{r^5 \delta / 4 - r^5 \delta / 2} \text{ for all } \delta \geq 4 \\ &\leq 2^{-r^5 \delta / 4} \end{aligned}$$

So $\text{Prob}[T_{S_r} > 39r^2 \cdot 72r^3(1 + \delta)] \leq 2^{-r^5 \delta / 4}$ for all $\delta \geq 4$. So the expected value of T_{S_r} , the construction time, is $O(r^5) = O(\log^{5/3}(n))$ with an exponentially decaying tail probability. \square

It is natural to ask whether faster construction of polymers is possible in non-deterministic systems: systems that do not construct a single terminal polymer. A two-monomer-type insertion system consisting of the initiator $(s_1, s_2)(s_2^*, s_1^*)$ and monomer types $(s_2^*, s_1^*, s_1, s_2)^+$, and $(s_2^*, x, x, s_2)^+$ simultaneously constructs polymers of all lengths $n \geq 3$ in expected time $O(\log n)$ via balanced insertion sequences of logarithmic length. Moreover, any polymer in any system has $\Omega(\log n)$ expected construction time, since every insertion takes $\Omega(1)$ expected time, and constructing a polymer of length n requires an insertion sequence of length at least $\lfloor \log_2(n-2) \rfloor$. So if assembling anything is permitted, then this two-monomer-type system is asymptotically optimal.

What about a milder form of non-determinism: permitting system to construct a finite total number of polymers? Our next result proves that even this relaxation is sufficient to improve construction time. The key idea is allow large sets of monomer types to “compete” to insert into a common insertion site first. This competition increases the total concentration of insertable monomer types, reducing the expected insertion time, but results in a non-deterministic system.

Theorem 4.3. *For any positive, odd integer r , there exists an insertion system constructing a finite set of polymers with $O(r^2)$ monomer types that constructs a polymer of length $n = 2^{\Theta(r^2)}$ in $O(\log^{3/2}(n))$ expected time. Moreover, the expected time has an exponentially decaying tail probability.*

Proof. We use a construction similar to one used in the proof of Theorem 4.2, but with a few key differences. First, this construction carries out the execution of a double for-loop, not a triple for-loop. Second, each increment step involves a sequence of “guesses”: non-deterministic insertions where many different monomer types can be inserted. Incorrect guesses stop further execution, while At these insertions, only one monomer type allows continued execution of the for-loop, and all others cause the loop to break.

We use a and b for the outer and inner for-loop variables, and a parity bit to encode the parity of b . Insertion sites of the form $(s_{f_p(a)}, s_{f_p(b)})(s_{f_p(b)}^*, s_{f_p(a)}^*)$ encode counter states, where $p = b \bmod 2$ and $f_i(n) = n + 2ir^2$. The parity bit is used to avoid an issue with incorrect guesses causing repeated insertion sites (and thus an infinite set of polymers). Three types of variable increments are carried out by the counter:

Inner even-to-odd: If $b < r$ and b is even, then $(s_{f_0(a)}, s_{f_0(b)})(s_{f_0(b)}^*, s_{f_0(a)}^*) \rightsquigarrow (s_{f_1(a)}, s_{f_1(b+1)})(s_{f_1(b+1)}^*, s_{f_1(a)}^*)$.

Inner odd-to-even: If $b < r$ and b is odd, then $(s_{f_1(a)}, s_{f_1(b)})(s_{f_1(b)}^*, s_{f_1(a)}^*) \rightsquigarrow (s_{f_0(a)}, s_{f_0(b+1)})(s_{f_0(b+1)}^*, s_{f_0(a)}^*)$.

Outer: If $b = r$ and $a < r$, then $(s_{f_1(a)}, s_{f_1(r)})(s_{f_1(r)}^*, s_{f_1(a)}^*) \rightsquigarrow (s_{f_0(a+1)}, s_{f_0(0)})(s_{f_0(0)}^*, s_{f_0(a+1)}^*)$.

As in the construction of Theorem 4.2, this counter carries out a sequence of $\Theta(r^2)$ insertions, which are used to construct a length $2^{\Theta(r^2)}$ polymer by simultaneously duplicating each site during an inner even-to-odd increment. A complete set of monomers for the system can be seen in Table 3.

In the remainder of the proof we detail the implementation of each increment type. The bulk of the effort lies in proving that every insertion site accepting multiple monomer types has a single *correct* monomer type that continues the execution of the loops, and all other *incorrect* monomer types yield insertion sites that do not accept any monomer types. We call such an insertion site *growth-deterministic*, and if every insertion site appearing in a system is growth deterministic, the polymer constructed by only inserting correct monomer types is the longest polymer constructed by the system.

Step	Inner even-to-odd monomer types ($b < r$, b even)	
1	$(s_{f_1(b+1)}^*, s_{f_0(a)}^*, s_{f_0(a)}, x)^-$	
2	$(s_{f_0(b)}^*, x, s_{f_1(a)}, s_{f_1(b+1)})^+$	$(s_{f_1(b+1)}^*, s_{f_1(a)}^*, s_{f_0(a)}, s_{f_2(b+2)})^-$
3	$(s_{f_2(b+2)}^*, s_{f_2(a)}^*, x, s_{f_1(b+1)})^+$	$(s_{f_0(b+2)}^*, s_{f_0(a)}^*, s_{f_2(a)}, x)^-$
3	$(s_{f_2(b+2)}^*, x, s_{f_3(a)}, s_{f_0(b+2)})^+$	$(x, s_{f_3(a)}^*, s_{f_0(a)}, s_{f_0(b+2)})^-$
Inner odd-to-even monomer types ($b < r$, b odd)		
1	$(s_{f_0(b+1)}^*, s_{f_1(a)}^*, s_{f_1(a)}, x)^-$	$(s_{f_1(b)}^*, x, s_{f_0(a)}, s_{f_0(b+1)})^+$
	$(s_{f_0(b+1)}^*, s_{f_0(a)}^*, s_{f_1(a)}, x)^-$	
Outer monomer types ($a < r$)		
1	$(s_{f_1(r)}^*, x, s_{f_3(a+1)}, s_{f_1(r)})^+$	$(s_{f_0(0)}^*, s_{f_3(a+1)}^*, s_{f_1(a)}, x)^-$
2	$(s_{f_1(r)}^*, x, s_{f_0(a+1)}, s_{f_0(0)})^+$	$(s_{f_0(0)}^*, s_{f_0(a+1)}^*, s_{f_3(a+1)}, x)^-$

Table 3: The set of all monomer types used to construct a monomer of size n in $O(\log^{3/2}(n))$ expected time while constructing only finite polymers.

Inner odd-to-even increment. We implement inner odd-to-even increments in a single step of three insertions. In an sequence of three correct insertions, the monomer types

$(s_{f_0(b+1)}^*, s_{f_1(a)}^*, s_{f_1(a)}, x)^- \quad (s_{f_1(b)}^*, x, s_{f_0(a)}, s_{f_0(b+1)})^+ \quad (s_{f_0(b+1)}^*, s_{f_0(a)}^*, s_{f_1(a)}, x)^-$
are used. This set of monomers results the sequence of insertions

$$\begin{aligned}
& (s_{f_1(a)}, s_{f_1(b)}) \diamond (s_{f_1(b)}^*, s_{f_1(a)}^*) \\
& (s_{f_1(a)}, s_{f_1(b)}) \diamond (s_{f_0(b+1)}^*, s_{f_1(a)}^*, s_{f_1(a)}, x)(s_{f_1(b)}^*, s_{f_1(a)}^*) \\
& (s_{f_1(a)}, s_{f_1(b)}) \diamond (s_{f_0(b+1)}^*, s_{f_1(a)}^*) \\
& (s_{f_1(a)}, s_{f_1(b)})(s_{f_1(b)}^*, x, s_{f_0(a)}, s_{f_0(b+1)}) \diamond (s_{f_0(b+1)}^*, s_{f_1(a)}^*) \\
& (s_{f_0(a)}, s_{f_0(b+1)}) \diamond (s_{f_0(b+1)}^*, s_{f_1(a)}^*) \\
& (s_{f_0(a)}, s_{f_0(b+1)}) \diamond (s_{f_0(b+1)}^*, s_{f_0(a)}^*, s_{f_1(a)}, x)(s_{f_0(b+1)}^*, s_{f_1(a)}^*) \\
& (s_{f_0(a)}, s_{f_0(b+1)}) \diamond (s_{f_0(b+1)}^*, s_{f_0(a)}^*)
\end{aligned}$$

The first insertion site $(s_{f_1(a)}, s_{f_1(b)})(s_{f_1(b)}^*, s_{f_1(a)}^*)$ accepts any monomer of the form $(_, s_{f_1(a)}^*, s_{f_1(a)}, _)^-$ or $(s_{f_1(b)}^*, _, _, s_{f_1(b)})^+$. The only such monomer types in the system are $(s_{f_0(i+1)}^*, s_{f_1(a)}^*, s_{f_1(a)}, x)^-$, where i is an odd integer with $0 \leq i < r$ (inner odd-to-even monomers). So the insertion has the form

$$\begin{aligned}
& (s_{f_1(a)}, s_{f_1(b)}) \diamond (s_{f_1(b)}^*, s_{f_1(a)}^*) \\
& (s_{f_1(a)}, s_{f_1(b)})(s_{f_0(i+1)}^*, s_{f_1(a)}^*, s_{f_1(a)}, x)(s_{f_1(b)}^*, s_{f_1(a)}^*)
\end{aligned}$$

Only the left resulting site is insertable. Moreover, this site is only insertable if $i = b$, since the only monomers of the form $(s_{f_1(b)}^*, _, _, s_{f_0(i+1)})^+$ in the system have $i = b$. So only one monomer type inserted into the first site enables further growth: $(s_{f_0(b+1)}^*, s_{f_1(a)}^*, s_{f_1(a)}, x)^-$. Thus the first site is growth-deterministic.

The second insertion site accepts monomers of the form $(s_{f_1(b)}^*, _, _, s_{f_0(b+1)})^+$. The only such monomer types in the system are $(s_{f_1(b)}^*, x, s_{f_0(i)}, s_{f_0(b+1)})^+$ with $0 \leq i \leq r$ (inner odd-to-even monomers). So the insertion has the form

$$(s_{f_1(a)}, s_{f_1(b)}) \diamond (s_{f_0(b+1)}^*, s_{f_1(a)}^*) \\ (s_{f_1(a)}, s_{f_1(b)})(s_{f_1(b)}^*, x, s_{f_0(i)}, s_{f_0(b+1)})(s_{f_0(b+1)}^*, s_{f_1(a)}^*)$$

Only the right resulting site is insertable. Moreover, this site is only insertable if $i = a$, since the only monomers of the form $(_, s_{f_0(i)}^*, s_{f_1(a)}, _)^-$ in the system have $i = a$. So only one monomer inserted into the second site enables further growth: $(s_{f_1(b)}^*, x, s_{f_0(a)}, s_{f_0(b+1)})^+$. So the second site is growth-deterministic.

The third insertion site accepts monomers of the form $(_, s_{f_0(a)}^*, s_{f_1(a)}, _)^-$. The only such monomer types in the system are $(s_{f_0(i+1)}^*, s_{f_0(a)}^*, s_{f_1(a)}, x)^-$ where i is an odd integer with $0 \leq i < r$ (inner odd-to-even monomers). So the insertion has the form

$$(s_{f_0(a)}, s_{f_0(b+1)}) \diamond (s_{f_0(b+1)}^*, s_{f_1(a)}^*) \\ (s_{f_0(a)}, s_{f_0(b+1)})(s_{f_0(i+1)}^*, s_{f_0(a)}^*, s_{f_1(a)}, x)(s_{f_0(b+1)}^*, s_{f_1(a)}^*)$$

Only the left resulting site is insertable. Moreover, this site is only insertable if $i = b$, since the only monomers of the form $(s_{f_0(b+1)}, _, _, s_{f_0(i+1)}^*)^+$ or $(_, s_{f_0(a)}^*, s_{f_0(a)}, _)^-$ in the system are the latter, which require $i = b$ to insert. So only one monomer inserted into the second site enables further growth: $(s_{f_0(b+1)}^*, s_{f_0(a)}^*, s_{f_1(a)}, x)^-$. So the third site is growth-deterministic.

Outer increment. We implement outer increments in two steps:

$$(s_{f_1(a)}, s_{f_1(r)})(s_{f_1(r)}^*, s_{f_1(a)}^*) \xrightarrow{1} (s_{f_3(a+1)}, s_{f_1(r)})(s_{f_0(0)}^*, s_{f_3(a+1)}^*) \\ (s_{f_3(a+1)}, s_{f_1(r)})(s_{f_0(0)}^*, s_{f_3(a+1)}^*) \xrightarrow{2} (s_{f_0(a+1)}, s_{f_0(0)})(s_{f_0(0)}^*, s_{f_0(a+1)}^*)$$

In a sequence of four correct insertions, the monomer types

Step 1: $(s_{f_1(r)}^*, x, s_{f_3(a+1)}, s_{f_1(r)})^+$ and $(s_{f_0(0)}^*, s_{f_3(a+1)}^*, s_{f_1(a)}, x)^-$.

Step 2: $(s_{f_1(r)}^*, x, s_{f_0(a+1)}, s_{f_0(0)})^+$ and $(s_{f_0(0)}^*, s_{f_0(a+1)}^*, s_{f_3(a+1)}, x)^-$.

are used. This set of monomers results in the sequence of insertions

$$(s_{f_1(a)}, s_{f_1(r)}) \diamond (s_{f_1(r)}^*, s_{f_1(a)}^*) \\ (s_{f_1(a)}, s_{f_1(r)})(s_{f_1(r)}^*, x, s_{f_3(a+1)}, s_{f_1(r)}) \diamond (s_{f_1(r)}^*, s_{f_1(a)}^*)$$

$$\begin{aligned}
& (s_{f_3(a+1)}, s_{f_1(r)}) \diamond (s_{f_1(r)}^*, s_{f_1(a)}^*) \\
& (s_{f_3(a+1)}, s_{f_1(r)}) \diamond (s_{f_0(0)}^*, s_{f_3(a+1)}^*, s_{f_1(a)}, x)(s_{f_1(r)}^*, s_{f_1(a)}^*) \\
& (s_{f_3(a+1)}, s_{f_1(r)}) \diamond (s_{f_0(0)}^*, s_{f_3(a+1)}^*) \\
& (s_{f_3(a+1)}, s_{f_1(r)})(s_{f_1(r)}^*, x, s_{f_0(a+1)}, s_{f_0(0)}) \diamond (s_{f_0(0)}^*, s_{f_3(a+1)}^*) \\
& (s_{f_0(a+1)}, s_{f_0(0)}) \diamond (s_{f_0(0)}^*, s_{f_3(a+1)}^*) \\
& (s_{f_0(a+1)}, s_{f_0(0)}) \diamond (s_{f_0(0)}^*, s_{f_3(a+1)}^*) \\
& (s_{f_0(a+1)}, s_{f_0(0)}) \diamond (s_{f_0(0)}^*, s_{f_3(a+1)}^*)
\end{aligned}$$

The first insertion site $(s_{f_1(a)}, s_{f_1(r)})(s_{f_1(r)}^*, s_{f_1(a)}^*)$ accepts any monomer of the form $(_, s_{f_1(a)}^*, s_{f_1(a)}, _)^-$ or $(s_{f_1(r)}^*, _, _, s_{f_1(r)})^+$. There are two such monomer types in the system: $(s_{f_0(i+1)}^*, s_{f_1(a)}^*, s_{f_1(a)}, x)^-$ where i is an odd integer with $0 \leq i < r$ (inner odd-to-even monomers) and $(s_{f_1(r)}^*, x, s_{f_3(i+1)}, s_{f_1(r)})^+$ with $0 \leq i < r$ (outer monomers). Insertion of the first type has the form

$$\begin{aligned}
& (s_{f_1(a)}, s_{f_1(r)}) \diamond (s_{f_1(r)}^*, s_{f_1(a)}^*) \\
& (s_{f_1(a)}, s_{f_1(r)})(s_{f_0(i+1)}^*, s_{f_1(a)}^*, s_{f_1(a)}, x)(s_{f_1(r)}^*, s_{f_1(a)}^*)
\end{aligned}$$

Neither resulting site is insertable, since the left site requires a monomer of the form $(s_{f_1(r)}^*, _, _, s_{f_0(i+1)})^+$ not found in the system. Insertion of the second type has the form

$$\begin{aligned}
& (s_{f_1(a)}, s_{f_1(r)}) \diamond (s_{f_1(r)}^*, s_{f_1(a)}^*) \\
& (s_{f_1(a)}, s_{f_1(r)})(s_{f_1(r)}^*, x, s_{f_3(i+1)}, s_{f_1(r)})(s_{f_1(r)}^*, s_{f_1(a)}^*)
\end{aligned}$$

Only the right resulting site is insertable. Moreover, this site is only insertable if $i = a$, since the only monomer type of the form $(_, s_{f_3(i+1)}^*, s_{f_1(a)}, _)^-$ has $i = a$. So only one monomer type inserted into the first site enables further growth: $(s_{f_1(r)}, x, s_{f_3(a+1)}, s_{f_1(r)})^+$. So the first insertion site is growth-deterministic.

The second insertion site $(s_{f_3(a+1)}, s_{f_1(r)})(s_{f_1(r)}^*, s_{f_1(a)}^*)$ accepts any monomer of the form $(_, s_{f_3(a+1)}^*, s_{f_1(a)}, _)^-$. The only such monomer type in the system is $(s_{f_0(0)}^*, s_{f_3(a+1)}^*, s_{f_1(a)}, x)^-$. So the second insertion site is deterministic and thus growth-deterministic.

The third insertion site $(s_{f_3(a+1)}, s_{f_1(r)})(s_{f_0(0)}^*, s_{f_3(a+1)}^*)$ accepts any monomer of the form $(s_{f_1(r)}^*, _, _, s_{f_0(0)})^+$. The only such monomer types in the system are $(s_{f_1(r)}, x, s_{f_0(i+1)}, s_{f_0(0)})^+$ with $0 \leq i < r$. So the insertion has the form

$$\begin{aligned}
& (s_{f_3(a+1)}, s_{f_1(r)}) \diamond (s_{f_0(0)}^*, s_{f_3(a+1)}^*) \\
& (s_{f_3(a+1)}, s_{f_1(r)})(s_{f_1(r)}, x, s_{f_0(i+1)}, s_{f_0(0)})(s_{f_0(0)}^*, s_{f_3(a+1)}^*)
\end{aligned}$$

Only the resulting right site is insertable. Moreover, this site is only insertable if $i = a$, since the only monomers of the form $(_, s_{f_0(i+1)}, s_{f_3(a+1)}, _)^-$ in the system have $i = a$. So only one monomer type inserted into the third site

enables further growth: $(s_{f_1(r)}, x, s_{f_0(a+1)}, s_{f_0(0)})^+$. So the third insertion site is growth-deterministic.

The fourth insertion site $(s_{f_0(a+1)}, s_{f_0(0)})(s_{f_0(0)}^*, s_{f_3(a+1)}^*)$ accepts any monomer of the form $(_, s_{f_0(a+1)}^*, s_{f_3(a+1)}, _)^-$. The only such monomer types in the system are $(s_{f_0(0)}^*, s_{f_0(i+1)}^*, s_{f_3(i+1)}, x)^-$ with $0 \leq i < r$. So the insertion has the form

$$(s_{f_0(a+1)}, s_{f_0(0)}) \diamond (s_{f_0(0)}^*, s_{f_3(a+1)}^*) \\ (s_{f_0(a+1)}, s_{f_0(0)})(s_{f_0(0)}^*, s_{f_0(i+1)}^*, s_{f_3(i+1)}, x)(s_{f_0(0)}^*, s_{f_3(a+1)}^*)$$

Only the resulting left site is insertable. Moreover, this site is only insertable if $i = a$, since the only monomers of the form $(_, s_{f_0(a+1)}^*, s_{f_0(i+1)}, _)^-$ or $(s_{f_0(0)}^*, _, _, s_{f_0(0)})^+$ have $i = a$. So only one monomer type inserted into the fourth site enables further growth: $(s_{f_0(0)}^*, s_{f_0(i+1)}^*, s_{f_3(i+1)}, x)^-$. So the fourth insertion site is growth-deterministic.

Inner even-to-odd increment. We implement inner even-to-odd increments in three steps. The first step is a sequence of three insertions:

$$(s_{f_0(a)}, s_{f_0(b)}) \diamond (s_{f_0(b)}^*, s_{f_0(a)}^*) \\ (s_{f_0(a)}, s_{f_0(b)}) \diamond (s_{f_1(b+1)}^*, s_{f_0(a)}^*, s_{f_0(a)}, x)(s_{f_0(b)}^*, s_{f_0(a)}^*) \\ (s_{f_0(a)}, s_{f_0(b)}) \diamond (s_{f_1(b+1)}^*, s_{f_0(a)}^*) \\ (s_{f_0(a)}, s_{f_0(b)})(s_{f_0(b)}^*, x, s_{f_1(a)}, s_{f_1(b+1)}) \diamond (s_{f_1(b+1)}^*, s_{f_0(a)}^*) \\ (s_{f_1(a)}, s_{f_1(b+1)}) \diamond (s_{f_1(b+1)}^*, s_{f_0(a)}^*) \\ (s_{f_1(a)}, s_{f_1(b+1)}) \diamond (s_{f_1(b+1)}^*, s_{f_1(a)}^*, s_{f_0(a)}, s_{f_2(b+2)}) \diamond (s_{f_1(b+1)}^*, s_{f_0(a)}^*)$$

This results in two insertion sites. The left insertion site is an incremented version of the original site, while the right site is not. The second step is used to modify the right site into an incremented version of the original site, but incremented *by 2*:

$$(s_{f_0(a)}, s_{f_2(b+2)}) \diamond (s_{f_1(b+1)}^*, s_{f_0(a)}^*) \\ (s_{f_0(a)}, s_{f_2(b+2)}) \diamond (s_{f_2(b+2)}^*, s_{f_2(a)}^*, x, s_{f_1(b+1)}) \diamond (s_{f_1(b+1)}^*, s_{f_0(a)}^*) \\ (s_{f_0(a)}, s_{f_2(b+2)}) \diamond (s_{f_2(b+2)}^*, s_{f_2(a)}^*) \\ (s_{f_0(a)}, s_{f_2(b+2)}) \diamond (s_{f_0(b+2)}^*, s_{f_0(a)}^*, s_{f_2(a)}, x)(s_{f_2(b+2)}^*, s_{f_2(a)}^*) \\ (s_{f_0(a)}, s_{f_2(b+2)}) \diamond (s_{f_0(b+2)}^*, s_{f_0(a)}^*) \\ (s_{f_0(a)}, s_{f_2(b+2)})(s_{f_2(b+2)}^*, x, s_{f_3(a)}, s_{f_0(b+2)}) \diamond (s_{f_0(b+2)}^*, s_{f_0(a)}^*) \\ (s_{f_3(a)}, s_{f_0(b+2)}) \diamond (s_{f_0(b+2)}^*, s_{f_0(a)}^*) \\ (s_{f_3(a)}, s_{f_0(b+2)}) \diamond (x, s_{f_3(a)}^*, s_{f_0(a)}, s_{f_0(b+2)}) \diamond (s_{f_0(b+2)}^*, s_{f_0(a)}^*) \\ (s_{f_0(a)}, s_{f_0(b+2)})(s_{f_0(b+2)}^*, s_{f_0(a)}^*)$$

In the case that $b + 2 > r$, the site is simply uninsertable.

The first site $(s_{f_0(a)}, s_{f_0(b)})(s_{f_0(b)}^*, s_{f_0(a)}^*)$ accepts any monomer of the form $(s_{f_0(b)}^*, _, _, s_{f_0(b)})^+$ or $(_, s_{f_0(a)}^*, s_{f_0(a)}, _)^-$. The only such monomer types in the system are $(s_{f_1(i+1)}^*, s_{f_0(a)}^*, s_{f_0(a)}, x)^-$ with $0 \leq i < r$. So the insertion has the form

$$(s_{f_0(a)}, s_{f_0(b)}) \diamond (s_{f_0(b)}^*, s_{f_0(a)}^*) \\ (s_{f_0(a)}, s_{f_0(b)})(s_{f_1(i+1)}^*, s_{f_0(a)}^*, s_{f_0(a)}, x)(s_{f_0(b)}^*, s_{f_0(a)}^*)$$

Only the resulting left site is insertable. Moreover, this site is only insertable if $i = b$, since the only monomers of the form $(s_{f_0(b)}^*, _, _, s_{f_1(i+1)})^+$ have $i = b$. So only one monomer type inserted into the first site enables further growth: $(s_{f_1(b+1)}^*, s_{f_0(a)}^*, s_{f_0(a)}, x)^+$. So the first insertion site is growth-deterministic.

The second site $(s_{f_0(a)}, s_{f_0(b)})(s_{f_1(b+1)}^*, s_{f_0(a)}^*)$ accepts any monomer of the form $(s_{f_0(b)}^*, _, _, s_{f_1(b+1)})^+$. The only such monomer types in the system are $(s_{f_0(b)}, x, s_{f_1(i)}, s_{f_1(b+1)})^+$ with $0 \leq i \leq r$. So the insertion has the form

$$(s_{f_0(a)}, s_{f_0(b)}) \diamond (s_{f_1(b+1)}^*, s_{f_0(a)}^*) \\ (s_{f_0(a)}, s_{f_0(b)})(s_{f_0(b)}, x, s_{f_1(i)}, s_{f_1(b+1)})(s_{f_1(b+1)}^*, s_{f_0(a)}^*)$$

Only the right resulting site is insertable. Moreover, this site is only insertable if $i = a$, since the only monomers of the form $(_, s_{f_1(i)}^*, s_{f_0(a)}, _)^-$ have $i = a$. So only one monomer type inserted into the second site enables further growth: $(s_{f_0(b)}, x, s_{f_1(a)}, s_{f_1(b+1)})^+$. So the second site is growth-deterministic.

The third site $(s_{f_1(a)}, s_{f_1(b+1)})(s_{f_1(b+1)}^*, s_{f_0(a)}^*)$ accepts any monomer of the form $(_, s_{f_1(a)}^*, s_{f_0(a)}, _)^-$. The only such monomer types in the system are $(s_{f_1(i+1)}^*, s_{f_1(a)}^*, s_{f_0(a)}, s_{f_2(i+2)})^-$ with $0 \leq i < r$. So the insertion has the form

$$(s_{f_1(a)}, s_{f_1(b+1)}) \diamond (s_{f_1(b+1)}^*, s_{f_0(a)}^*) \\ (s_{f_1(a)}, s_{f_1(b+1)})(s_{f_1(i+1)}^*, s_{f_1(a)}^*, s_{f_0(a)}, s_{f_2(i+2)})(s_{f_1(b+1)}^*, s_{f_0(a)}^*)$$

Both of the resulting sites are insertable. The left site is only insertable if $i = b$, since the only monomers of the form $(s_{f_1(b+1)}^*, _, _, s_{f_1(i+1)})^+$ or $(_, s_{f_1(a)}^*, s_{f_1(a)}, _)^-$ require $i = b$ to insert. So only one monomer type inserted into the third site enables further growth: $(s_{f_1(b+1)}^*, s_{f_1(a)}^*, s_{f_0(a)}, s_{f_2(b+2)})^-$.

The right site is also only insertable if $i = b$, since the only monomers of the form $(s_{f_2(i+2)}^*, _, _, s_{f_1(b+1)})^+$ have $i = b$. So only one monomer type inserted into the third site enables further growth: $(s_{f_1(b+1)}^*, s_{f_1(a)}^*, s_{f_0(a)}, s_{f_2(b+2)})^-$. So the third site is growth-deterministic.

The fourth site $(s_{f_0(a)}, s_{f_2(b+2)})(s_{f_1(b+1)}^*, s_{f_0(a)}^*)$ accepts any monomer of the form $(s_{f_2(b+2)}^*, _, _, s_{f_1(b+1)})^+$. The only such monomer types in the system are $(s_{f_2(b+2)}^*, s_{f_2(i)}^*, x, s_{f_1(b+1)})^+$ with $0 \leq i \leq r$. So the insertion has the form

$$(s_{f_0(a)}, s_{f_2(b+2)}) \diamond (s_{f_1(b+1)}^*, s_{f_0(a)}^*) \\ (s_{f_0(a)}, s_{f_2(b+2)})(s_{f_2(b+2)}^*, s_{f_2(i)}^*, x, s_{f_1(b+1)})(s_{f_1(b+1)}^*, s_{f_0(a)}^*)$$

Only the left resulting site is insertable. Moreover, this site is only insertable if $i = a$, since the only monomers of the form $(_, s_{f_0(a)}^*, s_{f_2(i)}, _)^-$ have $i = a$. So only one monomer type inserted into the fourth site enables further growth: $(s_{f_2(b+2)}^*, s_{f_2(a)}, x, s_{f_1(b+1)})^+$. So the fourth site is growth-deterministic.

The fifth site $(s_{f_0(a)}, s_{f_2(b+2)})(s_{f_2(b+2)}^*, s_{f_2(a)}^*)$ accepts any monomer of the form $(_, s_{f_0(a)}^*, s_{f_2(a)}, _)^-$. The only such monomer types in the system are $(s_{f_0(i+2)}^*, s_{f_0(a)}, s_{f_2(a)}, x)^-$ with $0 \leq i < r$. So the insertion has the form

$$(s_{f_0(a)}, s_{f_2(b+2)}) \diamond (s_{f_2(b+2)}^*, s_{f_2(a)}^*)$$

$$(s_{f_0(a)}, s_{f_2(b+2)})(s_{f_0(i+2)}^*, s_{f_0(a)}, s_{f_2(a)}, x)(s_{f_2(b+2)}^*, s_{f_2(a)}^*)$$

Only the left resulting site is insertable. Moreover, this site is only insertable if $i = b$, since the only monomers of the form $(s_{f_2(b+2)}^*, _, _, s_{f_0(i+2)})^+$ have $i = b$. So only one monomer type inserted into the fifth site enables further growth: $(s_{f_0(b+2)}^*, s_{f_0(a)}, s_{f_2(a)}, x)^-$. So the fifth site is growth-deterministic.

The sixth site $(s_{f_0(a)}, s_{f_2(b+2)})(s_{f_0(b+2)}^*, s_{f_0(a)}^*)$ accepts any monomer type of the form $(s_{f_2(b+2)}^*, _, _, s_{f_0(b+2)})^-$. The only such monomer types in the system are $(s_{f_2(b+2)}^*, x, s_{f_3(i)}, s_{f_0(b+2)})^+$ with $0 \leq i \leq r$. So the insertion has the form

$$(s_{f_0(a)}, s_{f_2(b+2)}) \diamond (s_{f_0(b+2)}^*, s_{f_0(a)}^*)$$

$$(s_{f_0(a)}, s_{f_2(b+2)})(s_{f_2(b+2)}^*, x, s_{f_3(i)}, s_{f_0(b+2)})(s_{f_0(b+2)}^*, s_{f_0(a)}^*)$$

Only the right resulting site is insertable. Moreover, this site is only insertable if $i = a$, since the only monomers of the form $(_, s_{f_3(i)}, s_{f_0(a)}, _)^-$ have $i = a$. So only one monomer type inserted into the sixth site enables further growth: $(x, s_{f_3(a)}, s_{f_0(a)}, s_{f_0(b+2)})^-$. So the sixth site is growth-deterministic.

The seventh site $(s_{f_3(a)}, s_{f_0(b+2)})(s_{f_0(b+2)}^*, s_{f_0(a)}^*)$ accepts any monomer type of the form $(_, s_{f_3(a)}^*, s_{f_0(a)}, _)^-$. The only such monomer types in the system are $(x, s_{f_3(a)}^*, s_{f_0(a)}, s_{f_0(i+2)})^-$ with $0 \leq i < r$. So the insertion has the form

$$(s_{f_3(a)}, s_{f_0(b+2)}) \diamond (s_{f_0(b+2)}^*, s_{f_0(a)}^*)$$

$$(s_{f_3(a)}, s_{f_0(b+2)})(x, s_{f_3(a)}^*, s_{f_0(a)}, s_{f_0(i+2)})(s_{f_0(b+2)}^*, s_{f_0(a)}^*)$$

Only the right resulting site is insertable. Moreover, this site is only insertable if $i = b$, since the only monomers of the form $(s_{f_0(i+2)}, _, _, s_{f_0(b+2)})^+$ or $(_, s_{f_0(a)}^*, s_{f_0(a)}, _)^-$ require $i = b$ to insert. So only one monomer type inserted into the seventh site enables further growth: $(x, s_{f_3(a)}^*, s_{f_0(a)}, s_{f_0(b+2)})^-$. So the seventh site is growth-deterministic.

Putting it together. Since all insertion sites are growth-deterministic, and the correct monomer types carry out the execution of a double for-loop, the system constructs only polymers of length at most the length of the polymer where all double for-loops are executed to completion. The size P_i of a sub-polymer with initiator encoding some value between 0 and $(r+1)^2 - 1$ can be bounded by $2P_{i+2} + 7 \leq P_i \leq 2P_{i+1} + 7$, since either $i+1$ or $i+2$ is an inner

even-to-odd increment step (inserting 7 monomers) and no step inserts more than 7 monomers. Moreover, $P_{(r+1)^2-2} > 0$. So $P_0 + 2$, the size of the longest terminal polymer of the system, is $2^{\Theta(r^2)}$.

Running time. Recall that the purpose of this construction was to achieve reduced running time by allowing multiple monomer types to “compete” for common sites. Each monomer form listed in Table 3 corresponds to a 1, r , or $r + 1$ sets of monomer types, with each set consisting of types insertable into common sites. For instance, the form $(s_{f_1(b+1)}^*, s_{f_0(a)}^*, s_{f_0(a)}, x)^-$ corresponds to $r + 1$ sets of monomer types, one for each value of a , where all types in a set can be inserted into exactly the sites of the form $(s_{f_0(a)}, c)(\bar{c}, s_{f_0(a)}^*)$.

We assign equal concentrations to each set of monomer types, with the concentrations of the types within the set distributed equally. There are $2 + 6r + 6(r + 1) \leq 2r + 6r + 12r = 20r$ total sets, so the total concentration of the types in each set is at least $1/(20r)$. We consider the expected construction time of the longest polymer of the system. The construction of this polymer is complete as soon as every counter’s variables have reached the value $a = b = r$, i.e. the insertion site has been modified to be $(s_{f_1(r)}, s_{f_1(r)})(s_{f_1(r)}^*, s_{f_1(r)}^*)$ and the monomer $(s_{f_1(r)}^*, x, s_{f_3(a+1)}, s_{f_1(r)})^+$ has been inserted. There are fewer than 2^{r^2} such insertions, and each insertion can occur once at most $7 \cdot (r + 1)^2 \leq 28r^2$ previous insertions have occurred.

So an upper bound on the expected time T_r for each such insertion is described as a sum of $28r^2$ random variables, each with expected time $20r$. The Chernoff bound for exponential random variables implies the following upper bound on T_r :

$$\begin{aligned} \text{Prob}[T_r > 20r \cdot 28r^2(1 + \delta)] &\leq e^{-20 \cdot 28r^3 \delta^2 / (2 + \delta)} \\ &\leq e^{-r^3 \delta^2 / (2 + \delta)} \\ &\leq e^{-r^3 \delta^2 / (2\delta)} \text{ for all } \delta \geq 2 \\ &\leq e^{-r^3 \delta / 2} \end{aligned}$$

Let T_{S_r} be the total running time of the system. Then we can bound T_{S_r} from above using the bound for T_r :

$$\begin{aligned} \text{Prob}[T_{S_r} > 20r \cdot 28r^2(1 + \delta)] &\leq 2^{r^2} \cdot e^{-r^3 \delta / 2} \\ &\leq 2^{r^2} 2^{-r^3 \delta / 2} \\ &\leq 2^{r^2 - r^3 \delta / 2} \\ &\leq 2^{r^3 \delta / 4 - r^3 \delta / 2} \text{ for all } \delta \geq 4 \\ &\leq 2^{-r^3 \delta / 4} \end{aligned}$$

So $\text{Prob}[T_{S_r} > 20r \cdot 28r^2(1 + \delta)] \leq 2^{-r^3 \delta / 4}$ for all $\delta \geq 4$. So the expected value of T_{S_r} , the construction time, is $O(r^3) = O(\log^{3/2}(n))$ with an exponentially decaying tail probability. \square \square

5 Negative Results for Polymer Growth

Here we show that the constructions in the previous section are asymptotically the best possible. This is done in Theorems 5.4, 5.6 and 5.8. A collection of intervening lemmas are used to prove bounds on the number of monomer types and expected time to carry out an *insertion sequence*: a sequence of monomer insertions where each insertion is into a site created by the previous insertion.

Observe that if two monomer types of the same sign are insertable into a common site, then the set of sites each can be inserted into is equal. Nearly all of the lemmas involve consideration of not only monomer types, but *insertion sets*: maximal sets of same-signed monomer types sharing a common set of insertion sites each can be inserted into. The system described in Theorem 4.2 uses $\Theta(\log^{2/3}(n))$ singleton insertion sets. On the other hand, the construction of Theorem 4.3 uses $\Theta(\sqrt{\log n})$ insertion sets, each of $\Theta(\sqrt{\log n})$ monomer types, e.g. $\{(s_{f_1(b+1)}^*, s_{f_0(a)}^*, s_{f_0(a)}, x)^- : 0 \leq b < r\}$ in the first row of Table 3. These larger sets decrease construction time by increasing the total concentration of monomer types insertable into each site.

The first several lemmas of the section are used to prove Theorem 5.4 and Lemma 5.5, a lemma describing the trade-off between the number of monomer types and expected construction time for systems constructing finite polymer sets. This lemma is combined with extremal bounds on the minimum number of monomer types and insertion sets to prove that the constructions of Theorem 4.2 and 4.3 are optimal at both ends of this trade-off.

Lemma 5.1. *Any insertion sequence of length l with no repeated insertion sites has $\Theta(l)$ sites of the form $(a, b)(\bar{c}, \bar{a})$ with $b \neq c$.*

Proof. Insertion sites have one of three forms:

Positive: $(a, b)(\bar{c}, \bar{a})$ with $b \neq c$.

Mixed: $(a, b)(\bar{b}, \bar{a})$.

Negative: $(b, a)(\bar{a}, \bar{c})$ with $b \neq c$.

We prove that every sequence of four consecutive insertion sites has at least one positive site. Consider such a sequence of four sites (and the three intervening insertions). If the first site is positive, we're done. If the first site is mixed, then the first monomer type inserted may be negative or positive. If a negative monomer type is inserted:

$$\begin{aligned} &(a, b) \diamond (\bar{b}, \bar{a}) \\ &(a, b)(\bar{c}, \bar{a}, a, d)(\bar{b}, \bar{a}) \end{aligned}$$

Since the sequence does not repeat sites, $b \neq c, d$ and either second site, $(a, b)(\bar{c}, \bar{a})$ or $(a, d)(\bar{b}, \bar{a})$, is positive. If a positive monomer type is inserted:

$$(a, b) \diamond (\bar{b}, \bar{a})$$

$$(a, b)(\bar{b}, \bar{c}, d, b)(\bar{b}, \bar{a})$$

As before, $a, d \neq c$ since sites cannot repeat. So the next insertion must use a negative monomer type. We assume the left site is used next in the sequence (a symmetric argument works if the right site is used instead). The entire insertion sequence has the form:

$$\begin{aligned} & (a, b) \diamond (\bar{b}, \bar{a}) \\ & (a, b) \diamond (\bar{b}, \bar{c}, d, b)(\bar{b}, \bar{a}) \\ & (a, b) \diamond (\bar{b}, \bar{c}) \\ & (a, b)(\bar{c}, \bar{a}, c, f)(\bar{b}, \bar{c}) \end{aligned}$$

As before, $e, f \neq b$ since sites cannot repeat. So the next site, either $(a, b)(\bar{c}, \bar{a})$ with $b \neq e$ or $(c, f)(\bar{b}, \bar{c})$ with $f \neq b$ is positive. So the third site in the sequence is positive. Finally, if the initial site is negative then the first monomer type inserted is negative:

$$\begin{aligned} & (b, a) \diamond (\bar{a}, \bar{c}) \\ & (b, a) \diamond (\bar{d}, \bar{b}, c, e)(\bar{a}, \bar{c}) \\ & (b, a) \diamond (\bar{d}, \bar{b}) \end{aligned}$$

We assume that the left site is used next in the sequence (a symmetric argument works if the right side is used instead). If $a \neq d$ then the second site is positive. Otherwise the second site is mixed, and by previous argument, at most two more insertions (a total of three) will take place until a positive site appears. So in the entire sequence of length l , a positive site appears at least once in every sequence of four consecutive sites. \square \square

Lemma 5.2. *Any insertion sequence with no repeated insertion sites using k monomer types forming m insertion sets has length $O(m\sqrt{k})$.*

Proof. Let $\mathcal{S} = (\Sigma, \Delta, Q, R)$ be the insertion system containing the sequence. Relabel the symbols in $\Sigma \cup \{s^* : s \in \Sigma\}$ as s_1, s_2, \dots, s_{4k} , with some of these symbols possibly unused. Note that this implies that for every s_i , $\bar{s}_i = s_j$ for some $j \in 1, 2, \dots, 4k$. Let l be the length of the sequence. By Lemma 5.1, $\Theta(l)$ sites are *positive*: they have the form $(s_a, s_b)(\bar{s}_c, \bar{s}_a)$ with $b \neq c$.

A bound of $\sum_{i=1}^{4k} \min(|L_i|, |R_i|) \leq 3m$. Let L_i and R_i be the sets of monomer types of the forms $(_, _, s_i, _)^\pm$ and $(_, \bar{s}_i, _, _)^\pm$, respectively, used in the insertion sequence. Each positive site $(s_i, s_b)(\bar{s}_c, \bar{s}_i)$ consists of a left monomer in L_i and right monomer in R_i . Every occurrence of a positive site in the sequence is followed by the use of the left or right resulting site, e.g.:

$$\begin{aligned} & (s_i, s_b) \diamond (\bar{s}_c, \bar{s}_i) \\ & (s_i, s_b) \diamond (\bar{s}_b, \bar{s}_d, s_e, s_c)(\bar{s}_c, \bar{s}_i) \\ & (s_i, s_b) \diamond (\bar{s}_b, \bar{s}_d) \end{aligned}$$

It is the case that d is unique for c , i.e. no two insertions into positive sites using the left resulting sites both use monomers of the form $(\overline{s_b}, \overline{s_d}, _, _)^+$, since such a pair of monomers implies the sequence repeats the insertion site $(s_a, s_b)(\overline{s_b}, \overline{s_d})$. A similar claim holds for e and b in the case that the right site is used. So inserting into the resulting site requires a monomer from a distinct insertion set $\{(_, \overline{s_i}, s_d, _)^- \in \Delta\}$ or, in the special case that $i = d$, $\{(\overline{s_b}, _, _, s_b)^+ \in \Delta\}$.

The resulting sites require monomers from a number of distinct insertion sets equal to the sum of two values. First, the number of times the left side is used with a distinct c and a monomer is inserted into a site $(s_i, s_b)(\overline{s_b}, \overline{s_d})$ with d unique for c . Second, the number of times the right side is used with a distinct b and a monomer is inserted into a site $(s_e, s_c)(\overline{s_c}, \overline{s_i})$ with e unique for b . An assignment of left and right side usage that minimizes the number of distinct insertion sets needed is nearly equivalent to a minimum vertex covering of the following bipartite graph:

- A node $L_{(i,b)}$ for every site $(s_i, s_b)(\overline{s_c}, \overline{s_i})$ in the insertion sequence.
- A node $R_{(c,i)}$ for every site $(s_i, s_b)(\overline{s_c}, \overline{s_i})$ in the insertion sequence.
- An edge $(L_{(i,b)}, R_{(c,i)})$ for every site $(s_i, s_b)(\overline{s_c}, \overline{s_i})$ in the insertion sequence.

Selecting a vertex to cover a given edge corresponds to using the resulting left or right site of the edge's site, e.g. selecting $R_{(c,i)}$ to cover the edge $(L_{(i,b)}, R_{(c,i)})$ corresponds to using the resulting left site and inserting a monomer type of the form $(_, \overline{s_i}, s_d, _)^-$, where d is unique for c . By König's theorem (see [7, 2]), since the graph is bipartite, the size of a minimum vertex covering is equal to the size of a maximum matching, which is bounded from above by $\sum_{i=1}^{4k} \min(|L_i|, |R_i|)$.

However, an insertion set $\{(_, \overline{s_e}, s_d, _)^- \in \Delta\}$ corresponds to selecting both $R_{(c,i)}$, where d is unique for c , and $L_{(j,b)}$, where e is unique for b . So the number of insertion sets needed may be as little as half the size of the vertex cover of the bipartite graph. Additionally, one site may not be inserted into. So $\sum_{i=1}^{4k} \min(|L_i|, |R_i|) - 1 \leq 2m$ and $\sum_{i=1}^{4k} \min(|L_i|, |R_i|) \leq 3m$.

Maximizing insertion sequence length. Consider the number of positive sites y accepting some monomer type. We proved that $\Omega(l) = y$ and it is easily observed that $y \leq \sum_{i=1}^{4k} \min(m, |L_i| \cdot |R_i|)$. We also proved that $\sum_{i=1}^{4k} \min(|L_i|, |R_i|) \leq 3m$ and it is easily observed that $\sum_{i=1}^{4k} \max(|L_i|, |R_i|) \leq 2k$, since each monomer type is in at most one L_i and one R_i . This gives the following set of constraints:

1. $\Omega(l) = \sum_{i=1}^{4k} \min(m, |L_i| \cdot |R_i|)$.
2. $\sum_{i=1}^{4k} \min(|L_i|, |R_i|) \leq 3m$.
3. $\sum_{i=1}^{4k} \max(|L_i|, |R_i|) \leq 2k$.

Observe that $|L_i| \cdot |R_i| = \min(|L_i|, |R_i|) \cdot \max(|L_i|, |R_i|)$. Define two new variables $y_i = \min(|L_i|, |R_i|)$ and $z_i = \max(|L_i|, |R_i|)$ for an alternate formulation of the previous constraints:

1. $\Omega(l) = \sum_{i=1}^{4k} \min(m, y_i z_i)$.
2. $\sum_{i=1}^{4k} y_i \leq 3m$.
3. $\sum_{i=1}^{4k} z_i \leq 2k$.

Relax y_i, z_i to be real-valued and let $W = \{i : y_i z_i > 0\}$. If $0 < y_i z_i, y_j z_j < m$ for some $i \neq j$ and $y_i = \max(y_i, z_i, y_j, z_j)$, then $\min(m, y_i z_i) + \min(m, y_j z_j) < \min(m, y_i(z_i + \varepsilon)) + \min(m, y_j(z_j - \varepsilon))$ for sufficiently small $\varepsilon > 0$. More generally, if $0 < y_i z_i, y_j z_j < m$ for some $i \neq j$ then the values of y_i, z_i, y_j, z_j can be modified to increase $\sum_{i=1}^{4k} \min(m, y_i z_i)$. Therefore the maximum value is achieved when $m = y_i z_i$ for all but at most one $i \in W$.

We claim that it cannot be that $y_i z_i = m$ for $6\sqrt{k}$ distinct values of i . By contradiction, assume so. So $|W| \geq 6\sqrt{k}$ and the average value of y_i for $i \in W$ must be less than $3m/(6\sqrt{k}) = m/(2\sqrt{k})$. So for a subset $W' \subseteq W$ with $|W'| \geq |W|/2 \geq 3\sqrt{k}$, $y_i \leq 2 \cdot m/(2\sqrt{k}) = m/\sqrt{k}$ for all $i \in W'$. For every $i \in W'$, because $y_i \leq m/\sqrt{k}$ and $y_i z_i = m$, it must be the case that $z_i \geq \sqrt{k}$. So $\sum_{i=1}^{4k} z_i \geq |W'| \cdot \sqrt{k} \geq 3k$, a contradiction with the constraint that $\sum_{i=1}^{4k} z_i \leq 2k$.

So the maximum value is achieved when $m = y_i z_i$ for all but at most one $i \in W$, with $|W| + 1 < 6\sqrt{k} + 1 < 7\sqrt{k}$. So $\sum_{i=1}^{4k} \min(m, y_i z_i) \leq (|W| + 1)m < 7m\sqrt{k}$. So $\Omega(l) = 7m\sqrt{k}$ and $l = O(m\sqrt{k})$. \square \square

Lemma 5.3. *An insertion sequence of length l using monomer types from m insertion sets with no repeated insertion sites takes $\Omega(ml)$ expected time.*

Proof. By linearity of expectation, the total expected time of the insertions is equal to the sum of the expected time for each insertion. By Lemma 5.1, $\Theta(l)$ sites are both *positive*, i.e. they have the form $(s_a, s_b)(\overline{s_c}, \overline{s_a})$ with $b \neq c$, and accept the monomer types of a positive, non-empty insertion set.

Let m be the number of insertion sets formed by the monomer types inserted into these $\Omega(l)$ sites. Let c_1, c_2, \dots, c_m be the sums of the concentrations of the monomer types in these sets, and x_1, x_2, \dots, x_m be the number of times a monomer from each set is inserted in the subsequence. Then the total expected time for all of the insertions in the subsequence is $\sum_{i=1}^m x_i/c_i$. Moreover, these variables are subject to the following constraints:

1. $\sum_{i=1}^m x_i = \Omega(l)$ (total number of insertions is $\Omega(l)$).
2. $\sum_{i=1}^m c_i \leq 1$ (total concentration is at most 1).

Minimizing expected time. We now consider minimizing the total expected time subject to these constraints, starting with proving that $x_i/c_i =$

x_j/c_j for all $1 \leq i, j \leq m$. That is, that the ratio of the number of sites that accept an insertion set to the total concentrations of the monomer types in the set is equal for all sets. Assume, without loss of generality, that $x_i/c_i > x_j/c_j$ and $c_i, c_j > 0$. Then it can be shown algebraically that the following two statements hold:

1. If $c_j \geq c_i$, then for sufficiently small $\varepsilon > 0$, $\frac{x_i}{c_i} + \frac{x_j}{c_j} > \frac{x_i}{c_i + \varepsilon} + \frac{x_j}{c_j - \varepsilon}$.
2. If $c_j < c_i$, then for sufficiently small $\varepsilon > 0$, $\frac{x_i}{c_i} + \frac{x_j}{c_j} > \frac{x_i}{c_i - \varepsilon} + \frac{x_j}{c_j + \varepsilon}$.

Since the ratios of every pair of monomer types are equal,

$$\frac{c_i}{1} \leq \frac{c_i}{\sum_{i=1}^m c_i} = \frac{x_i}{\sum_{i=1}^m x_i} = O(x_i/l)$$

So $\Omega(l) = x_i/c_i$ and $\Omega(ml) = \sum_{i=1}^m x_i/c_i$. □ □

We now combine these two lemmas to prove a lower bound for deterministic systems matching the construction of Theorem 4.2 and a trade-off lower bound for systems constructing a finite set of polymers used to prove that the constructions of Theorems 4.2 and 4.3 are optimal at both ends of the length-speed trade-off curve.

Theorem 5.4. *Any polymer of length n constructed by a deterministic insertion system with k monomer types takes $\Omega(\log^{5/3}(n))$ expected time.*

Proof. Suppose a system $\mathcal{S} = (\Sigma, \Delta, Q, R)$ has an insertion set with monomer types m_1 and m_2 , and inserts m_1 into some polymer. Then all polymers constructed by systems $(\Sigma, \Delta - \{m_1\}, Q, R)$ and $(\Sigma, \Delta - \{m_2\}, Q, R)$ are constructed by \mathcal{S} and both have polymers not constructed by the other containing m_2 and m_1 , respectively. So any deterministic system with no unused monomer types has exclusively singleton insertion sets, i.e. $m = k$. By Lemma 5.2, $n = 2^{O(k\sqrt{k})}$ and $m = k = \Omega(\log^{2/3}(n))$. So by Lemma 5.3, the expected time to construct a polymer of length n is $\Omega(ml) = \Omega(\log^{5/3}(n))$. □

Lemma 5.5. *Any polymer of length n constructed by an insertion system with k monomer types constructing a finite set of polymers takes $\Omega(\log^2(n)/\sqrt{k})$ expected time.*

Proof. By Lemma 5.2, $n = 2^{O(m\sqrt{k})}$. So $m = \Omega(\log n/\sqrt{k})$. Constructing any polymer of length n requires an insertion system of length $l = \Omega(\log n)$. Then by Lemma 5.3, the expected time to construct any polymer of length n is $\Omega(ml) = \Omega(\log^2(n)/\sqrt{k})$. □

Now Lemma 5.5 is combined with additional bounds on the minimum values of k and m to prove that the constructions in Section 4 are optimal in both monomer types and expected construction time for systems that construct a finite set of polymers and a polymer using the fewest monomer types (Theorem 5.6) and the least expected time (Theorem 5.8).

Theorem 5.6. *Any polymer constructed by an insertion system with k monomer types constructing a finite set of polymers has length $2^{O(k^{3/2})}$. Moreover, constructing a polymer of length $n = 2^{\Theta(k^{3/2})}$ takes $\Omega(\log^{5/3}(n))$ expected time.*

Proof. First, observe that constructing any polymer of length n in a system constructing a finite set of polymers involves an insertion sequence of length at least $\log_2(n)$ with no repeated insertion sites. By Lemma 5.2, $\log_2(n) = O(m\sqrt{k})$, where m and k are the number of monomer types used in the insertion sequence. Since the number of insertion sets m is at most k (if each monomer type forms a singleton set), $m \leq k$ and $\log_2(n) = O(k\sqrt{k}) = O(k^{3/2})$. So $n = 2^{\Theta(k^{3/2})}$ and $k = \Theta(\log^{2/3}(n))$. By Lemma 5.5, such a polymer requires $\Omega(\log^2(n)/\sqrt{\log^{2/3}(n)}) = \Omega(\log^{5/3}(n))$ expected construction time. \square \square

Before proving an expected-time lower bound for all systems constructing finite polymer sets, we prove a helpful lemma showing that the number of insertion sets cannot be too much smaller than the number of monomer types:

Lemma 5.7. *Any insertion sequence of length l with no repeated insertion sites using k monomer types forming m insertion sets has $m = \Omega(\sqrt{k})$.*

Proof. Notice that this bound can only be obtained by assuming the monomer types are used to carry out an insertion sequence, since it is possible to have an arbitrarily large set of monomer types belonging to a single insertion set. The number of monomer types used is at most the length of the insertion sequence ($k \leq l$), and the remainder of the proof is spent proving that the number of insertion sites in a system with m insertion sets is $O(m^2)$ ($l = O(m^2)$), giving the desired inequality.

Let $\mathcal{S} = (\Sigma, \Delta, Q, R)$ be the insertion system containing the sequence. Relabel the symbols in $\Sigma \cup \{s^* : s \in \Sigma\}$ as s_1, s_2, \dots, s_{4k} , with some of these symbols possibly unused. By Lemma 5.1, $\Omega(l)$ sites are *positive*: they have the form $(s_a, s_b)(s_c, \overline{s_a})$ with $b \neq c$.

Since the second monomer inserted to create the site must be negative, each positive site consists of at least one negative monomer type. Let L_i^- and R_i^- be the sets of monomer types of the forms $(_, _, s_i, _)^-$ and $(_, \overline{s_i}, _, _)^-$, respectively, used in the insertion sequence of length l . For a specific i , there exists a site of the form $(s_i, s_b)(s_c, \overline{s_i})$ only if $|L_i^-| + |R_i^-| > 0$. So the number of values of i such that a site of the form $(s_i, s_b)(s_c, \overline{s_i})$ exists is at most $\sum_{i=1}^{4k} |L_i^-| + \sum_{i=1}^{4k} |R_i^-|$. Since all monomer types of a negative insertion set belong to the same L_i^- and R_i^- , $\sum_{i=1}^{4k} |L_i^-| + \sum_{i=1}^{4k} |R_i^-| \leq 2m$.

Next, observe there are at most m sites of the form $(s_i, s_b)(s_c, \overline{s_i})$ that accept a monomer, since each site requires a monomer from a different positive insertion set. So the total number of positive sites that accept a monomer is at most $2m \cdot m = 2m^2$. Since there are $\Omega(l)$ positive sites in the insertion sequence, $\Omega(l) = 2m^2$ and $l = O(m^2)$. \square \square

Theorem 5.8. *Any polymer of length n constructed by an insertion system constructing a finite set of polymers takes $\Omega(\log^{3/2}(n))$ expected construction*

time. Moreover, constructing a polymer of length n in $\Theta(\log^{3/2}(n))$ expected time requires using $\Omega(\log n)$ monomer types.

Proof. First, observe that constructing a polymer of length n in a system constructing a finite set of polymers involves an insertion sequence of length $\log_2(n) \leq l$ with no repeated sites. By Lemmas 5.2 and 5.7, $\log_2(n) = O(m^2)$ and so $m = \Omega(\sqrt{\log n})$. Then by Lemma 5.3, carrying out the insertion sequence and completing the construction of the polymer takes $\Omega(ml) = \Omega(\log^{3/2}(n))$ expected time and by Lemma 5.5, $k = \Omega(\log n)$. \square \square

6 Open Problems

The results of Sections 4 and 5 describe the landscape of efficient polymer construction using insertion systems. Trivial systems of just a few polymers can construct polymers of arbitrary length in optimal time, but with the caveat that the growth is uncontrolled and the systems construct infinite set of polymers. On the other hand, deterministically constructing a polymer of length n requires $\Omega(\log^{2/3}(n))$ monomer types and $\Omega(\log^{5/3}(n))$ expected time, and both of these are achievable simultaneously. The intermediate situation of constructing finite sets of polymers is more intricate – polymers can be constructed faster, but with the trade-off of using more monomer types *and* non-determinism.

In our system achieving $O(\log^{3/2}(n))$ expected construction time (Theorem 4.3), an exponential number ($2^{\Theta(n \log \log n)}$) of “junk” terminal polymers are constructed. Since achieving such speed requires significantly fewer insertion sets than monomer types, some junk is necessary – but how much? One approach to proving a lower bound is to prove that insertion sites accepting large insertion sets imply a large number of terminal polymers. We have been unable to prove such an implication even in the simplest case:

Conjecture 6.1. *Every deterministic system with no unused monomer types has exclusively singleton insertion sets.*

Since assembling a polymer in $o(\log^{5/3}(n))$ expected time requires $\Omega(\log n)$ insertions along most insertion sequences are non-deterministic, the previous conjecture implies that any improvement in speed comes with an exponential number of junk terminal polymers:

Conjecture 6.2. *Any system constructing a polymer of length n in $O(\log^{3/2}(n))$ expected time constructs a set of $2^{\Omega(n)}$ polymers.*

Setting aside non-determinism, the trade-off between monomer types and construction time has a lower bound (Lemma 5.5) with matching upper bounds only at the extremes. Does there exist a parameterized system matching the lower bound across the entire range?

Conjecture 6.3. *For every combination of n and k such that $\log_2^{2/3}(n) \leq k \leq \log_2(n)$, there exists a system with k monomer types that constructs a polymer of length n in $O(\log^2 n / \sqrt{k})$ time.*

In Section 3, we were able to prove that insertion systems are expressively equivalent to context-free grammars. What minimal changes to the definition of insertion systems are necessary to achieve a Turing-universal model? For instance, what if we allow monomers to not only insert between existing monomers in a polymer, but replace or “kick out” an existing monomer or sequence of l monomers? Then we estimate a Chomsky-like hierarchy of expressive power for these “ l -insertion systems”:

Conjecture 6.4. *The expressive power of 1-insertion systems is equal to context-sensitive grammars.*

Conjecture 6.5. *The expressive power of l -insertion systems for $l \geq 2$ is equal to Turing machines.*

References

- [1] L. Adleman, Q. Cheng, A. Goel, and M.-D. Huang. Running time and program size for self-assembled squares. In *Proceedings of 33rd ACM Symposium on Theory of Computing (STOC)*, 2001.
- [2] J. A. Bondy and U. S. R. Murty. *Graph Theory with Applications*. Elsevier, New York, 1976.
- [3] H.-L. Chen, D. Doty, D. Holden, C. Thachuk, D. Woods, and C.-T. Yang. Fast algorithmic self-assembly of simple shapes using random agitation. In S. Murata and S. Kobayashi, editors, *DNA Computing and Molecular Programming*, volume 8727 of *LNCS*, pages 20–36. Springer Berlin Heidelberg, 2014.
- [4] M. Chen, D. Xin, and D. Woods. Parallel computation using active self-assembly. In D. Soloveichik and B. Yurke, editors, *DNA Computing and Molecular Programming*, volume 8141 of *LNCS*, pages 16–30. Springer Berlin Heidelberg, 2013.
- [5] N. Dabby. *Synthetic molecular machines for active self-assembly : prototype algorithms, designs, and experimental study*. PhD thesis, Caltech, 2013.
- [6] N. Dabby and H.-L. Chen. Active self-assembly of simple units using an insertion primitive. In *Proceedings of 24th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1526–1536, 2013.
- [7] R. Diestel. *Graph Theory*. Springer, Berlin, 2005.
- [8] D. Doty, J. H. Lutz, M. J. Patitz, R. T. Schweller, S. M. Summers, and D. Woods. The tile assembly model is intrinsically universal. In *Proceedings of 53rd IEEE Symposium on Foundations of Computer Sciences (FOCS)*, pages 302–310, 2012.

- [9] V. K. Gautam, P. C. Haddow, and M. Kuiper. Reliable self-assembly by self-triggered activation of enveloped DNA tiles. In A.-H. Dediu, C. Martín-Vide, B. Truthe, and M. A. Vega-Rodríguez, editors, *Theory and Practice of Natural Computing*, volume 8273 of *LNCS*, pages 68–79. Springer Berlin Heidelberg, 2013.
- [10] J. Hendricks, J. E. Padilla, M. J. Patitz, and T. A. Rogers. Signal transmission across tile assemblies: 3D static tiles simulate active self-assembly by 2D signal-passing tiles. In D. Soloveichik and B. Yurke, editors, *DNA Computing and Molecular Programming*, volume 8141 of *LNCS*, pages 90–104. Springer Berlin Heidelberg, 2013.
- [11] N. Jonoska and D. Karpenko. Active tile self-assembly, part 1: universality at temperature 1. *International Journal of Foundations of Computer Science*, 25(2):141–163, 2014.
- [12] N. Jonoska and D. Karpenko. Active tile self-assembly, part 2: self-similar structures and structural recursion. *International Journal of Foundations of Computer Science*, 25(2):165–194, 2014.
- [13] A. Keenan, R. Schweller, M. Sherman, and X. Zhong. Fast arithmetic in algorithmic self-assembly. Technical report, arXiv, 2013.
- [14] A. Keenan, R. Schweller, and X. Zhong. Exponential replication of patterns in the signal tile assembly model. In D. Soloveichik and B. Yurke, editors, *DNA Computing and Molecular Programming*, volume 8141 of *LNCS*, pages 118–132. Springer Berlin Heidelberg, 2013.
- [15] E. Klavins. Universal self-replication using graph grammars. In *Proceedings of International Conference on MEMS, NANO, and Smart Systems*, pages 198–204, 2004.
- [16] E. Klavins, R. Ghrist, and D. Lipsky. Graph grammars for self assembling robotic systems. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, volume 5, pages 5293–5300, 2004.
- [17] U. Majumder, T. H. LaBean, and J. H. Reif. Activatable tiles: Compact, robust programmable assembly and other applications. In M. H. Garzon and H. Yan, editors, *DNA Computing and Molecular Programming*, volume 4848 of *LNCS*, pages 15–25. Springer Berlin Heidelberg, 2008.
- [18] C. Malchik and A. Winslow. Tight bounds for active self-assembly using an insertion primitive. In *Proceedings of 22nd European Symposium on Algorithms (ESA)*, pages 677–688, 2014.
- [19] J. E. Padilla, W. Liu, and N. C. Seeman. Hierarchical self assembly of patterns from the robinson tilings: DNA tile design in an enhanced tile assembly model. *Natural Computing*, 11(2):323–338, 2012.

- [20] J. E. Padilla, M. J. Patitz, R. T. Schweller, N. C. Seeman, S. M. Summers, and X. Zhong. Asynchronous signal passing for tile self-assembly: fuel efficient computation and efficient assembly of shapes. *International Journal of Foundations of Computer Science*, 25(4):459–488, 2014.
- [21] P. W. K. Rothmund and E. Winfree. The program-size complexity of self-assembled squares (extended abstract). In *Proceedings of 32nd ACM Symposium on Theory of Computing (STOC)*, pages 459–468, 2000.
- [22] D. Soloveichik and E. Winfree. Complexity of self-assembled shapes. *SIAM Journal on Computing*, 36(6):1544–1569, 2007.
- [23] E. Winfree. *Algorithmic Self-Assembly of DNA*. PhD thesis, Caltech, 1998.
- [24] D. Woods, H.-L. Chen, S. Goodfriend, N. Dabby, E. Winfree, and P. Yin. Active self-assembly of algorithmic shapes and patterns in polylogarithmic time. In *Proceedings of 4th Conference on Innovations in Theoretical Computer Science (ITCS)*, pages 353–354, 2013.
- [25] D. Woods, H.-L. Chen, S. Goodfriend, N. Dabby, E. Winfree, and P. Yin. Active self-assembly of algorithmic shapes and patterns in polylogarithmic time. Technical report, arXiv, 2013.